2008

Large time-stepping methods for higher order time-dependent evolution equations

Xiaoliang Xie
Iowa State University

Large time-stepping methods for higher order time-dependent evolution equations

by

Xiaoliang Xie

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Applied Mathematics

Program of Study Committee:
Hailiang Liu, Major Professor
L. Steven Hou
Paul Sacks
Jim W. Evans
Sunder Sethuraman

Iowa State University

Ames, Iowa

2008

DEDICATION

To my dear Mother Ruilin Zhang and father XiXi Xie

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to Professor Hailiang Liu, my academic advisor, for his guidance and support. Throughout my PhD period, he often encouraged me to keep moving forward and to look for creative solutions. He always provided valuable comments when they were needed. His keen scientific insights were crucial for for the success of this dissertation, and they will always be a source of inspiration for me. Also, I want to thank Dr.Liu for his great help in the thesis revision process. If nothing else, he help me to have faith in myself even at the most difficult moments. I am sure he could not even fathom how much I have learned from him. Besides just math he has also taught me how to become a successful person. I really am so lucky to have had Dr.Hailiang Liu as my advisor.

In addition, I would like to thank my other committee members: Professors Paul Sacks, Sunder Sethuraman, L. Steven Hou and Jim Evans.

Besides, cooperation from Dr.Liu's other students are also important to my work. I appreciated all the support from Dr.Zhongming Wang and Dr.Haseena Ahmed.

Lastly, I am grateful to anyone and everyone who has had a hand in making my five years at Iowa State University such a success!

ABSTRACT

Our goal is to design large time step numerical schemes for a class of Partial Differential Equations. We propose a preconditioning approximation model $\partial_t u^\epsilon = \phi, \pounds(\epsilon, D)\phi = N$ for PDEs of the form $\partial_t u = L(u) + N(u), u(0, x) = u(x)$, where $\pounds$ is a linear operator and N is a nonlinear operator. While applying to linear equations, $\pounds$ can be chosen based on linear stability analysis. Nonlinear problem has to rely on certain intrinsic energy principle. In this paper we will give stability analysis for different types of linear equations and discrete energy analysis for some nonlinear equations. Also numerical examples together with some error analysis for both linear and nonlinear equations will be exhibited to illustrate the computational efficiency of the method.

CHAPTER 1. Introduction

Recently there has been a rise in research interest in the study of large time-stepping numerical methods for different types of Partial Differential Equations arising in various applications. Many of the underlying Partial Differential Equations are characterized by time dependence, high order derivatives and strong nonlinearity. Their solutions are expected to possess special properties such as pattern formation, something is not easily observed in a short period of time. Therefore, in solving such PDEs, long time behavior of numerial solutions is a critical issue.

We consider a class of PDEs of the form

$$\partial_t u = L(u) + N(u), \tag{1.1}$$

where $L = L(x, u, Du, D^2u, ...)$ is a linear operator that contains high order spatial derivatives and $N = N(x, u, Du, D^2u, ...)$ is a nonlinear operator that may also contain high order derivatives. The high order derivatives force small time step in explicit time-discretization according to the CFL condition. Various large time-stepping numerical schemes have been developed for these type of PDEs. The implicit-explicit scheme is to smartly combine the implicit and explicit approximation for linear and nonlinear spatial derivative operators. The stability property can be found in [4]. It has been proposed for reaction-diffusion problems [30], Navier-Stokes equations [20], and the KdV equation [6, 22]. The integrating factor IF method has been developed by Trefethen [33], and Cox and Matthews [9]. The idea is to solve high order linear operator exactly by making change of variables. An extension of implicit-explicit method which is called fast spectral algorithm has been developed by Fornberg and Driscoll [12] for purely dispersive equations and has been generalized by Driscoll [11] by using Runge-Kutta time-stepping. The main idea is to use different numerical scheme for the low,

medium and high wavenumbers separately. Another scheme is the exponential time differencing (ETD) scheme which has been developed by Cox and Matthews for stiff systems [9]. The ETD scheme applies the same integrating factor as in the IF approach, and the difference in the integration is over a single time step of length h. Later ETD schemes have been improved to fix the numerical inaccuracy due to cancellation errors, see [21].

1.1 Goals and Main Results

Our goal is to design a class of large time-stepping stable methods for different types of higher order PDEs. These will be developed gradually as the complexity of equations increases.

Part I: Linear equations(N=0 in (1.1))

Our key idea for design of large time-stepping numerical schemes is based on a preconditioning approximation

$$\partial_t u^\epsilon = \phi, \quad \pounds\phi = P(\partial_x)u, \tag{1.2}$$

where the preconditioning operator $\pounds(\epsilon, D)$ is selected to make $L(u)$ ϕ share equal regularity with u. In order for the approximate system to be accurate, we require the following:

(i)Consistency $\lim_{\epsilon\to 0} \pounds(\epsilon, D) = I$;

(ii)$\pounds(\epsilon, D)$ is a linear operator;

(iii)$\epsilon > 0$ is a parameter proportional to the time step to be determined.

We summarize the choice of $\pounds$ for three different types of linear operator L in Theorem (2.1.1).

1) $\pounds(\epsilon, \partial_x) = I - \epsilon P(\partial_x)$,$\epsilon = \frac{\Delta t}{2}$ for $Re(P(i\xi)) = 0$, for all $\xi \in \mathbb{R}$,

2) $\pounds(\epsilon, \partial_x) = I - \epsilon P(\partial_x)$ with $\frac{\Delta t}{2} \le \epsilon \le \Delta t$ for $Re(P(i\xi) < 0$, for all $\xi \in \mathbb{R}$,

3) $\pounds(\epsilon, \partial_x) = I - \epsilon(P(\partial_x) - c)$ with $\frac{\Delta t}{2} \le \epsilon \le \Delta t$ for $Re(P(i\xi)) \le c(c > 0)$, for all $\xi \in \mathbb{R}$.

Where $P(i\xi)$ is the Fourier transform of $P(\partial_x)$. By choosing $\pounds$ in this way, a semi-discrete scheme (2.4) is stable in the sense of

$$||u^{n+1}||_2 \le e^{c\Delta t}||u^n||_2. \tag{1.3}$$

We also discussed the effect of spatial discretization on the stability of the fully discrete scheme. We use A to denote the matrix that corresponds to the spacial operator $P(\partial_x)$ after finite difference discretization. We have the following results:
1) When $P(i\xi) \leq 0$ for $\xi \in \mathbb{R}$ and $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t$, a symmetric negative definite matrix A will make fully discretized scheme unconditionally stable in the sense of $||u^{n+1}|| \leq ||u^n||$.
2) When $P(i\xi)$ is pure imaginary and $\epsilon = \frac{\Delta t}{2}$, a skew-Hermitian matrix A will make fully discretized scheme unconditionally stable in the sense of $||u^{n+1}|| = ||u^n||$.
Where A is the spatial discretization matrix corresponding to linear operator $P(\partial_x)$.

Our study is mainly focused on stability issues, and we know that for linear equations, as long as we have a stable scheme it is also a convergent scheme due to the Lax Equivalence Theorem. Through the study of linear equations, we can build a solid foundation for nonlinear equations.

Part II: Semi-linear equations. When both N and L exist and the high order derivatives appear on the linear operator and the nonlinear operator is of low order, we regroup the operators and split the linear and nonlinear parts, applying the result for linear operator L according to the discussion for linear equations. Then we will deal with the nonlinear term wisely to get large time-stepping numerical schemes. Different nonlinear operator N should be treated differently to get a stable numerical scheme. We give three examples of nonlinear equations of this type, the Korteweg-de Vries, the Swift-Hohenberg equation and the generalized Kuramoto-Sivashinsky equation.

The KdV equation

For the KdV equation $u_t + uu_x + u_{xxx} = 0$, first we apply the operator $\pounds$ discussed in part I, then we use the Strang splitting technique, and apply 1/3 dispersive scheme on uu_x to

obtain:

$$\begin{aligned}
\frac{u_j^{(1)}-u_j^n}{\Delta t/2}+D_0D_+D_-\frac{u_j^{(1)}+u_j^n}{2} &= 0,\\
\frac{u_j^{(2)}-u_j^{(1)}}{\Delta t}+\frac{1}{6\Delta x}(u_{j+1}^*+u_j^*+u_{j-1}^*)(u_{j+1}^*-u_{j-1}^*) &= 0, u_j^*=\frac{u_j^{(2)}+u_j^{(1)}}{2},\\
\frac{u_j^{n+1}-u_j^{(2)}}{\Delta t/2}+D_0D_+D_-\frac{u_j^{n+1}+u_j^{(2)}}{2} &= 0, j=1,2,...,N,
\end{aligned} \tag{1.4}$$

where N is the grid point value, $u^{(1)}$ and $u^{(2)}$ are N dimensional vectors, Δt is the time step, Δx is the length of the partition, and $D_0D_+D_-u_j^n=\frac{u_{j+2}^n-2u_{j+1}^n+2u_{j-1}^n-u_{j-2}^n}{2h^3}$.

The nice feature of this scheme is to preserve both mass and energy at discrete level, i.e. $\sum_j(u_j^{n+1})^2=\sum_j(u_j^n)^2$, and $\sum_j(u_j^{n+1})=\sum_j(u_j^n)$. Meanwhile it is second order in both time and space. In a recent work [10], a second order finite volume method was introduced in order to preserve both mass and energy evolving two equations of both u and u^2. The scheme (1.4) shares the same properties as in [10], and is much easier to implement. Several numerical tests are performed by scheme (1.4)to show the accuracy, stability and robustness of long-time simulation.

The Swift-Hohenberg type equations

We consider the Swift-Hohenberg type equation

$$u_t=Lu+\frac{dF}{du}, \tag{1.5}$$

where $L=-\Delta^2-2\Delta$ is linear operator and F is a nonlinear function of u. The equation (1.5) admits a nonincreasing energy functional

$$\Psi=\int_\Omega\{\frac{1}{2}(\Delta u)^2-|\nabla u|^2-F(u)\}dxdy, \tag{1.6}$$

subject to the Dirichlet boundary condition

$$u=\Delta u=0,(x,y)\in\partial\Omega. \tag{1.7}$$

We apply the result in part I on linear operator L and use the divided difference technique on F to obtain

$$\frac{u^{n+1}-u^n}{\Delta t} = L\Big(\theta u^{n+1} + (1-\theta)u^n\Big) + \frac{F(u^{n+1})-F(u^n)}{u^{n+1}-u^n}. \tag{1.8}$$

We summarize the result of how to choose θ in semi-discrete scheme (1.8) to make it stable in the following:

1).If $\theta = \frac{1}{2}$, the scheme (5.6) is stable in the sense of

$$\Psi[u^{n+1}] = \Psi[u^n] - \int_\Omega \frac{(u^{n+1}-u^n)^2}{\Delta t} dxdy \le \Psi[u^n]. \tag{1.9}$$

for arbitrary Δt.

2). If $\theta > \frac{1}{2}$, then the scheme (5.6) is stable in the sense of

$$\Psi[u^{n+1}] \le \Psi[u^n],$$

for all $\Delta t \le \frac{1}{2\theta-1}$.

The numerical result is consistent with the theoretical result (5.1.1), and in our numerical tests we find the Dirichlet boundary condition $u = u_{xx} = 0$ is essential to capture the pattern formation for 1-D Swift-Hohenberg equation.

The Kuramoto-Sivashinsky equations

For the generalized Kuramoto-Sivashinsky equation

$$u_t + uu_x + (a(u)u_x)_x + (b'(u)(b(u)_x)_x)_x + (c(u)u_{xx})_{xx} = 0, x \in \mathbb{R}, \tag{1.10}$$

where a,b,c are continuous functions of u, we establish the L_2 stability of the finite difference scheme

$$\begin{aligned}\frac{u_j^{n+1}-u_j^n}{\Delta t} + \frac{1}{6h}(v_{j+1}+v_j+v_{j-1})(v_{j+1}-v_{j-1}) + D_+(a(v_j)D_-v_j) \\ +D_0(b'(v_j)D_+D_-b(v_j)) + D_+D_-(c(v_j)D_+D_-v_j) = 0,\end{aligned} \tag{1.11}$$

where $v = \frac{u^{n+1}+u^n}{2}$. Numerical tests for three different types of Kuramoto-Sivashinsky equations are conducted to show the accuracy and long time stability of this scheme.

Part III: Full nonlinear equations. Next we consider the case both N and L in (1.1) exist and the high order derivative appears on the nonlinear operator N. There has been significant research interest in these kinds of equations. In [34], efficient numerical schemes with particular emphasis on the use of large time steps have been developed for epitaxial growth models of thin films. In [14], a large time-stepping method for the Cahn-Hilliard equation has been developed and analyzed.

We consider the model equation

$$u_t = \Delta u - \nabla \cdot (\frac{\nabla u}{|\nabla u|}), x \in \Omega \in \mathbb{R}^2, \tag{1.12}$$

which is the gradient flow of the functional

$$I[u] = \frac{1}{2}\int_\Omega (|\nabla u| - 1)^2 dx. \tag{1.13}$$

We develope two schemes which are stable for large time steps in the sense of keeping the energy identity of continuous model (1.12) at the discrete level. The implementation of (7.10) in one dimension give us the desired solution $|\nabla u| = 1$.

1.2 Thesis Organization

This paper is organized as follows: in chapter 2, we will construct operator L for linear equations, guided by Von Neumann stability analysis, and we will discuss the spatial discretization for linear equations. We will also discuss the effect of spatial discretization on the stability of the fully discrete scheme in this section. In chapter 3, some numerical examples for linear equations are given. In chapter 4, we develop a second order large time-stepping numerical scheme for the Korteweg-de Vries equation, preserving both momentum and energy at the discrete level. Both theoretical and numerical results are given in this section. In chapter 5, we developed a numerical scheme for Swift-Hohenberg equation which preserves the nonincreasing energy functional at the discrete level. The pattern formation for $1-D$ Swift-Hohenberg

equation is displayed in the numerical test. In chapter 6, a stable numerical scheme will be developed for generalized Kuramoto-Sivashinsky equations. In chapter 7, we will develop two stable numerical schemes for a nonlinear model equation $u_t = \Delta u - \nabla \cdot (\frac{\nabla u}{|\nabla u|})$.

CHAPTER 2. Construction of the operator L

2.1 Construction of the operator L

In this chater, we discuss the selection of the smoothing operator for linear PDEs with constant coefficients. For simplicity, we only consider one-dimensional case. Similar analysis can be well applied to multidimensional case. We consider the linear problem with constant coefficients

$$\partial_t u = P(\partial_x)u, \quad u(x,0) = u_0(x), \quad x \in \mathbb{R}, \tag{2.1}$$

where $P(\partial_x) = \sum_{l=0}^{r} A_l \frac{\partial^l}{\partial x^l}$, A_l are constants.

We assume this problem is well-posed. To be self-contained, we recall the fundamental definition of well-posedness for problem (2.1), see ([13]).

Definition 2.1.1. The Cauchy problem (2.1) is well posed if , for every $u_0(x) \in C^\infty(\mathbb{R})$:
1.There exists a unique solution $u(x,t)$.
2.There are constants c and K, independent of $u_0(x)$, such that

$$||u(\cdot,t)||_2 \leq Ke^{ct}||u(.,0)||_2, \tag{2.2}$$

where $||.||_2$ denotes the norm in $L^2(\mathbb{R})$ space and $C^\infty(x,t)$ denotes the infinitely differentiable function space.

Our interest is to develop a large time step stable scheme to compute the solution of (2.1). To do so we first propose an approximate model to (2.1),

$$\begin{cases} \partial_t U = \phi, U = e^{-ct}u, \\ \pounds\phi = (P(\partial_x) - c)U, \end{cases} \tag{2.3}$$

where $\pounds(\epsilon, \partial_x) = I - \epsilon(P(\partial_x) - c)$ and the choice of ϵ depends on the operator $P(\partial_x)$.

We then discretize (2.3) in time by forward Euler method to obtain

$$\frac{U^{n+1} - U^n}{\Delta t} = \phi^n, \pounds\phi^n = (P(\partial_x) - c)U^n, \tag{2.4}$$

where $U^n = U^n(x) \approx U(t_n, x)$.

We now summarize the choice of ϵ in the following theorem.

Theorem 2.1.1. *The scheme (2.4) is stable in the sense of*

$$||u^{n+1}||_2 \leq e^{c\Delta t}||u^n||_2 \tag{2.5}$$

for any $\Delta t > 0$, if

1) $\pounds(\epsilon, \partial_x) = I - \frac{\Delta t}{2}P(\partial_x), c = 0$ for $Re(P(i\xi)) = 0$, for all $\xi \in \mathbb{R}$,

2) $\pounds(\epsilon, \partial_x) = I - \epsilon P(\partial_x)$ with $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t, c = 0$ for $Re(P(i\xi) < 0$, for all $\xi \in \mathbb{R}$,

3) $\pounds(\epsilon, \partial_x) = I - \epsilon(P(\partial_x) - c)$ with $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t$ for $Re(P(i\xi)) \leq c(c > 0)$, for all $\xi \in \mathbb{R}$.

Proof. In order to prove (2.5), it suffices to show

$$||U^{n+1}||_2 \leq ||U^n||_2.$$

We apply the Fourier transform on (2.4) to obtain

$$\hat{U}^{n+1}(\xi) = \hat{U}^n(\xi) + \Delta t\hat{\phi}^n(\xi), \quad \pounds(\epsilon, i\xi)\hat{\phi}^n(\xi) = (P(i\xi) - c)\hat{U}^n(\xi), \tag{2.6}$$

which gives

$$\hat{U}^{n+1} = Q(\xi)\hat{U}^n(\xi), \quad Q(\xi) = \Big(1 + \frac{(P(i\xi) - c)\Delta t}{\pounds(\epsilon, i\xi)}\Big). \tag{2.7}$$

Here $Q(\xi)$ plays the role of the amplification factor.

By taking $\pounds(\epsilon, \partial_x) = 1 - \epsilon(P(\partial_x) - c)$ in (2.7) we obtain

$$Q = \frac{1 - (\epsilon - \Delta t)(P(i\xi) - c)}{1 - \epsilon(P(i\xi) - c)}. \tag{2.8}$$

We now estimate Q for different cases.

1) With the choice $\epsilon = \frac{\Delta t}{2}$ and c=0,

$$Q = \frac{1 + \frac{\Delta t}{2} P(i\xi)}{1 - \frac{\Delta t}{2} P(i\xi)}. \tag{2.9}$$

From (2.9) and $Re(P(i\xi)) = 0$, it follows $|Q| = 1$, which guarantees the identity $||u^{n+1}||_2 = ||u^n||_2$.

2) From (2.8) together with the conditions $Re[P(i\xi)] \leq 0$, c=0 and $\epsilon \geq \frac{\Delta t}{2}$, we have

$$\begin{aligned} |Q| &= \Big|\frac{1 - (\epsilon - \Delta t)P(i\xi)}{1 - \epsilon P(i\xi)}\Big| \\ &= \frac{\sqrt{\big(1 - (\epsilon - \Delta t)Re(P(i\xi))\big)^2 + (Im(P(i\xi)))^2}}{\sqrt{\big(1 - \epsilon Re(P(i\xi))\big)^2 + (Im(P(i\xi)))^2}} \leq 1, \end{aligned}$$

where $Im(P(i\xi))$ denotes the imaginary part of $P(i\xi)$. The fact that $|Q| \leq 1$ implies $||u^{n+1}|| \leq ||u^n||$.

3)By the assumption $c > 0, Re(P(i\xi)) \leq 0$ we have $Re(P(i\xi)) - c \leq 0$. We substitute $P(i\xi) - c$ for $P(i\xi)$ in 2) to obtain $|Q| < 1$, which implies

$$||U^{n+1}|| \leq ||U^n||.$$

Then we substitute back $u = Ue^{ct}$ to obtain

$$\begin{aligned} ||u^{n+1}e^{-c(n+1)\Delta t}|| &\leq ||u^n e^{-cn\Delta t}||, \\ ||u^{n+1}|| &\leq ||u^n||e^{c\Delta t}. \end{aligned}$$

We complete the proof of the theorem.

2.2 Spatial Discretization

The semidiscrete scheme discussed in section (2.1) is unconditionally stable for any $\Delta t > 0$, in this section we discuss the effect of spatial discretization on the stability of the fully discrete scheme. For simplicity we consider c=0, then (2.4) can be rewritten as

$$\pounds(\epsilon, \partial_x)\frac{u^{n+1}(x) - u^n(x)}{\Delta t} = P(\partial_x)u^n(x), \tag{2.10}$$

where $\pounds(\epsilon, \partial_x)$ depends on the spatial operator $P(\partial_x)$. We discretize the spatial operator $P(\partial_x)$ by finite difference operator, we will investigate what additional conditions need to be imposed in order to keep the fully discrete scheme unconditionally stable.

2.2.1 Dissipative Case

When $P(\partial_x)$ contains only even order derivatives and $P(i\xi) \leq 0$ for any $\xi \in \mathbb{R}$, the equation is dissipative, from the discussion in section 2 we choose $\pounds = (I - \epsilon P(\partial_x))$ with $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t$, then (2.10) becomes

$$(I - \epsilon P(\partial_x))u^{n+1}(x) = (I - (\epsilon - \Delta t)P(\partial_x))u^n(x). \tag{2.11}$$

We apply a spatial discretization on (2.11) to obtain

$$(I - \epsilon A)u^{n+1} = (I - (\epsilon - \Delta t)A)u^n, \tag{2.12}$$

where $u^n = (u_1^n, u_2^n, \cdots, u_N^n)$ is an N dimensional vector, N is the number of grid points, A is a matrix derived from the discretization of $P(\partial_x)$.

For example for $P(\partial_x) = \partial_{xxxx}$ we use central difference to approximate the operator ∂_{xxxx},

$$\partial_{xxxx}u_i = \frac{u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}}{\Delta x^4},$$

where u_i is the i^{th} grid point value. If we use the periodic boundary condition then the matrix A corresponding to ∂_{xxxx} is

$$\mathbf{A} = \begin{pmatrix} 6 & -4 & 1 & \dots & 0 & 1 & -4 & 0 \\ -4 & 6 & -4 & 1 & \dots & 0 & 1 & 0 \\ 1 & -4 & 6 & -4 & 1 & \dots & 0 & 0 \\ 0 & 1 & -4 & 6 & -4 & 1 & 0 & \dots \\ & & \ddots & \ddots & \ddots & \ddots & & \\ 0 & 1 & \dots & 0 & 1 & -4 & 6 & -4 \\ 0 & -4 & 1 & \dots & 0 & 1 & -4 & 6 \end{pmatrix}.$$

Next we will show that the fully discrete scheme (2.12) remains stable for arbitrary time step for a class of A.

Lemma 2.2.1. *If A is symmetric and negative definite, and $(I-\epsilon A)$ is invertible, then scheme (2.12) is unconditionally stable in the sense of $||u^{n+1}|| \leq ||u^n||$, provided $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t$.*

Proof. For given A, $(I - \epsilon A)$ is invertible, then

$$u^{n+1} = (I - \epsilon A)^{-1}(I - (\epsilon - \Delta t)A)u^n.$$

Let $B = (I-\epsilon A)^{-1}(I-(\epsilon-\Delta t)A)$, and λ be an eigenvalue of B, associated with the eigenvector x, then

$$Bx = \lambda x.$$

That is

$$(I - \epsilon A)^{-1}(I - (\epsilon - \Delta t)A)x = \lambda x.$$

This leads to

$$Ax = -\frac{\lambda - 1}{\epsilon - \Delta t - \lambda\epsilon}x,$$

which implies

$$\lambda(A) = -\frac{\lambda(B) - 1}{\epsilon - \Delta t - \lambda(B)\epsilon}.$$

We know A is symmetric, negative definite, so its eigenvalues are negative.

$$-\frac{\lambda(B) - 1}{\epsilon - \Delta t - \lambda(B)\epsilon} < 0,$$

i.e.

$$((\lambda(B) - 1)\epsilon + \Delta t)(\lambda(B) - 1) < 0.$$

Hence

$$-1 \leq 1 - \frac{\Delta t}{\epsilon} \leq \lambda(B) \leq 1.$$

Note that $1 - \frac{\Delta t}{\epsilon} \geq -1$, because of $\epsilon \geq \frac{\Delta t}{2}$.

Since A is symmetric, so is B. It follows from linear algebra that B is unitarily equivalent to a diagonal matrix Λ with each diagonal element $\lambda_i \leqq 1, 0 \leq i \leq n$ where n is the dimension of Λ. Since we can write

$$u^{n+1} = U^T \Lambda U u^n,$$

it follows that

$$Uu^{n+1} = \Lambda U u^n.$$

Hence

$$||u^{n+1}||_2^2 = ||Uu^{n+1}||_2^2 \leqq ||Uu^n||_2^2 = ||u^n||_2^2,$$

which completes the proof.

2.2.2 Dispersive Case

When $P(\partial_x)$ contains only odd order derivatives, the equation is dispersive. In this case $\mathcal{L} = (I - \frac{\Delta t}{2} P(\partial_x))$, and (2.10) becomes

$$\Big(I - \frac{\Delta t}{2} P(\partial_x)\Big) u^{n+1}(x) = \Big(I + \frac{\Delta t}{2} P(\partial_x)\Big) u^n(x). \tag{2.13}$$

Assume A is the matrix corresponding to $P(\partial_x)$ and we discretize (2.13) in space to obtain

$$\Big(I - \frac{\Delta t}{2} A\Big) u^{n+1} = \Big(I + \frac{\Delta t}{2} A\Big) u^n. \tag{2.14}$$

For example $P(\partial_x) = \partial_{xxx}$, we use central difference to approximate the operator ∂_{xxx},

$$\partial_{xxx} u_i = \frac{u_{i+2} - 2u_{i+1} + 2u_{i-1} - u_{i-2}}{2\Delta x^3},$$

where u_i is the i^{th} grid point value. we use periodic boundary condition, then the matrix A corresponding to ∂_{xxx} is

$$\mathbf{A} = \begin{pmatrix} 0 & -2 & 1 & \dots & 0 & -1 & 2 & 0 \\ 2 & 0 & -2 & 1 & \dots & 0 & -1 & 0 \\ -1 & 2 & 0 & -2 & 1 & \dots & 0 & 0 \\ 0 & -1 & 2 & 0 & -2 & 1 & 0 & \dots \\ & & \ddots & \ddots & \ddots & \ddots & & \\ 0 & 1 & 0 & \dots & -1 & 2 & 0 & -2 \\ 0 & -2 & 1 & \dots & 0 & -1 & 2 & 0 \end{pmatrix},$$

which is skew-Hermitian. Then we have the following result

Lemma 2.2.2. *If A is skew-Hermitian, i.e.$A^* = -A$, and $(I - \epsilon A)$ is invertible, then scheme (2.14) is unconditionally stable in the sense that $||u^{n+1}||_{L_2} = ||u^n||_{L_2}$, where $||u^n||_{L_2} = \sqrt{\sum_{i=1}^{N}(u_i^n)^2 \Delta x}$.*

Proof. By the assumption that $(I - \epsilon A)$ is invertible, and from (2.14) we have

$$u^{n+1} = (I - \frac{\Delta t}{2}A)^{-1}(I + \frac{\Delta t}{2}A)u^n.$$

Let $B = (I - \frac{\Delta t}{2}A)$, so $B^* = (I + \frac{\Delta t}{2}A)$, then we have

$$||u^{n+1}||_{L_2}^2 = (B^{-1}B^*)^* B^{-1}B^* ||u^n||_{L_2}^2.$$

We notice that

$$(B^{-1}B^*)^* B^{-1}B^* = B(B^{-1})^* B^{-1}B^* = B(B^*)^{-1}B^{-1}B^* = B(BB^*)^{-1}B^*.$$

From the assumption A is skew-Hermitian, we have B is skew-Hermitian, so B is normal.

Then we have

$$B(BB^*)^{-1}B^* = B(B^*B)^{-1}B^* = BB^{-1}(B^*)^{-1}B^* = I,$$

i.e. $||u^{n+1}||_{L_2} = ||u^n||_{L_2}$. □

CHAPTER 3. Linear Equations

3.1 Diffusion equation

We consider a fourth order diffusion equation

$$u_t + u_{xxxx} = 0. \tag{3.1}$$

According to our discussion in Section 2, we choose $\pounds = 1 - \epsilon P(\partial_x)$ with $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t$, and we use central difference to approximate the spacial operator u_{xxxx} to obtain

$$(1 + \epsilon(D_+D_-)^2)u_j^{n+1} = (1 - (1-\epsilon)(D_+D_-)^2)u_j^n,$$

where $(D_+D_-)^2u_j^n = (u_{j+2}^n - 4u_{j+1}^n + 6u_j^n - 4u_{j-1}^n + u_{j-2}^n)/(\Delta x^4)$.

In our numerical test we choose the initial condition to be $u(x,0) = cos(2\pi x)$, $x \in [0,1]$, and the exact solution can be computed as $u_{exact}(x,t) = exp(-16\pi^4 t)cos(2\pi x)$.
First we perform a large time step test, we take large time steps $dt = 0.1, 0.2, 0.3, \cdots, 1$, and fix the number of grid points to be 41, the final time $tmax = 10$. From Figure 3.1 we see that, even when the time step is large, we still get a reasonable error, which shows that the scheme is stable for large time steps. Next we perform a test to see how the ratio $\theta = \frac{\epsilon}{\Delta t}$ effects the L_2 error. We fix the grid cell number to be 41, from our numerical experiments we find the number of grid points does not effect the shape of the graph. From Figure 3.2 we can conclude when Δt is small, the optimal θ should be close to $1/2$, and when Δt is large, the optimal θ should be close to 1.

We also perform a convergence test for the same equation. We fix the ending time to be $tmax = 0.01$ and take $\theta = 5/9$, and calculate the error and order of convergence by the

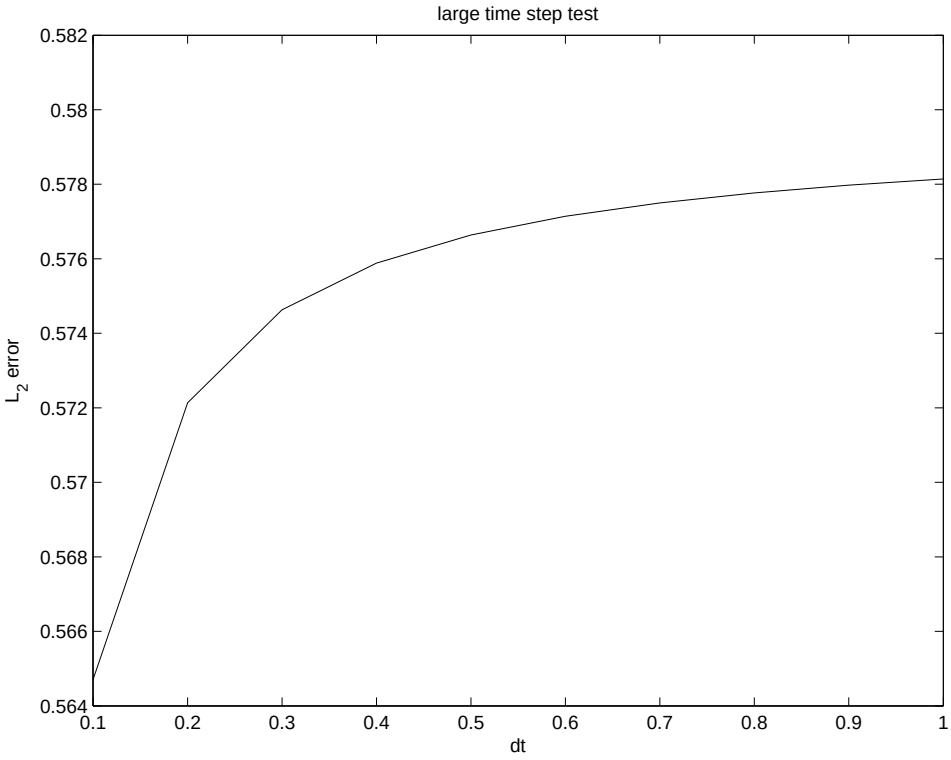

Figure 3.1: Large time step test $t = 10$, 41 cells.

following formulae

$$E_{k,l} = \sqrt{\sum (u_i^N - u_{exact}(x_i, t_N))^2}, \tag{3.2}$$

$$(r1)_{k,l} = \frac{1}{\log(2)} \log(\frac{E_{k,l+1} - E_{k,l}}{E_{k,l} - E_{k,l-1}}), \tag{3.3}$$

$$(r2)_{k,l} = \frac{1}{\log(2)} \log(\frac{E_{k,l} - E_{k-1,l}}{E_{k+1,l} - E_{k,l}}). \tag{3.4}$$

Table 3.1: L_2 error by computing equation (3.1)

N Δt	0.000001	0.000002	0.000004	0.000008
41	0.831E-08	0.851E-08	0.885E-08	0.943E-08
81	0.213E-08	0.23E-08	0.263E-08	0.325E-08
161	0.065E-08	0.081E-08	0.114E-08	0.176E-08
321	0.0291E-08	0.0447E-08	0.077E-08	0.139E-08

These three tables show the numerical scheme has convergence rate order 1 in time direction and order 2 in space direction.

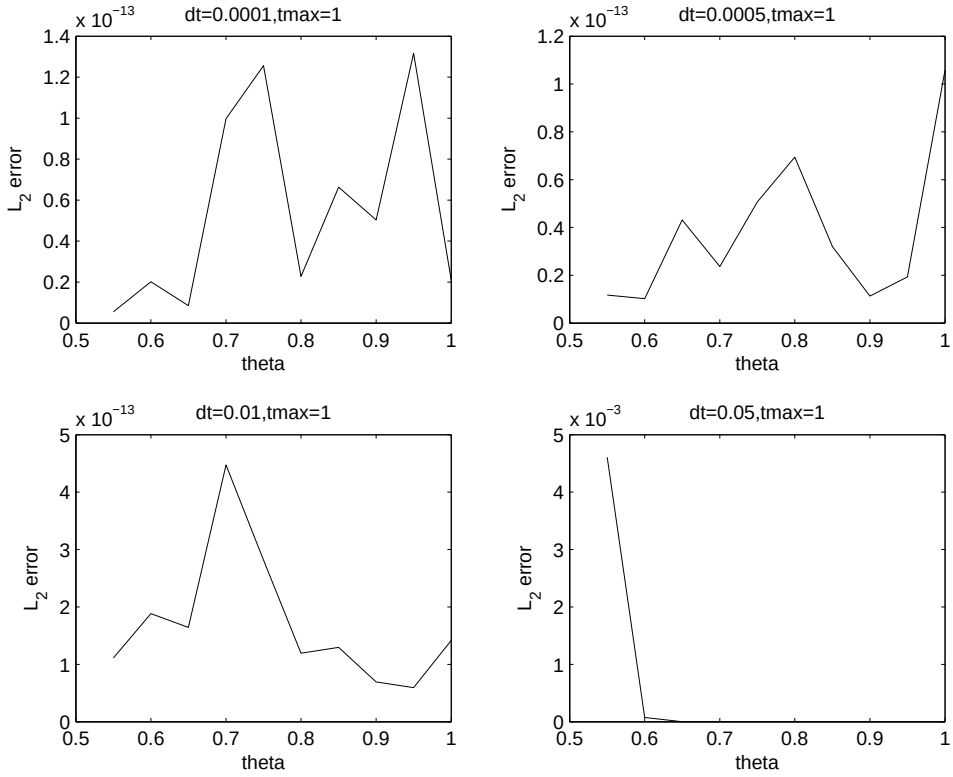

Figure 3.2: Small time step test $t = 1$, 41 cells.

3.2 Dispersive equation

We consider the dispersive equation $u_t + u_x + u_{xxx} = 0$, in which $P(\partial_x) = -\partial_x - \partial_{xxx}$. According to previous discussion we choose $\pounds = I + \frac{\Delta t}{2}(\partial_x + \partial_{xxx})$. The corresponding semi-

Table 3.2: order of convergence in the t direction $(r1)_{kl}$ of equation (3.1)

N Δt	0.000001	0.000002	0.000004	0.000008
41	-	0.9	0.73	-
81	-	0.97	0.9	-
161	-	0.98	0.94	-
321	-	1.06	0.94	-

Table 3.3: order of convergence in the x direction $(r2)_{k,l}$ of equation (3.1)

N Δt	0.000001	0.000002	0.000004	0.000008
41	-	-	-	-
81	2.06	2.06	2.06	2.06
161	2.05	2.02	2.02	2.01
321	-	-	-	-

discrete scheme is

$$(I + \frac{\Delta t}{2}(\partial_x + \partial_{xxx}))\frac{u^{n+1} - u^n}{\Delta t} = -u_x - u_{xxx},$$
$$(I + \frac{\Delta t}{2}(\partial_x + \partial_{xxx}))u^{n+1} = (I - \frac{\Delta t}{2}(\partial_x + \partial_{xxx}))u^n.$$

We use central difference for spatial discretization to obtain the numerical scheme

$$(1 + \frac{\Delta t}{2}(D_0 + D_0 D_+ D_-))u^{n+1} = (1 - \frac{\Delta t}{2}(D_0 + D_0 D_+ D_-))u^n. \tag{3.5}$$

We now carry out numerical test for equation (3.5) with the initial condition

$$u(x, t = 0) = \cos(2\pi x), x \in [0, 1].$$

The exact solution is

$$u_{exact}(x, t) = \cos(2\pi x - (2\pi - (2\pi)^3)t).$$

We display the numerical result at final time $tmax = 0.1$ in the following tables which show the scheme is second order in both time and space.

Table 3.4: L_2 Errors to compute dispersive equation

N Δt	0.00001	0.00002	0.00004	0.00008
21	0.4334	0.4334	0.4336	0.4340
41	0.1079	0.1080	0.1081	0.1085
81	0.0268	0.0268	0.0269	0.0274
161	0.0067	0.0067	0.0068	0.0072

Table 3.5: order of convergence in the t direction of dispersive equation $(r1)_{kl}$

N Δt	0.000001	0.000002	0.000004	0.000008
21	-	3.9609	1.5248	-
41	-	2.6141	1.7991	-
81	-	2.2240	1.9205	-
161	-	2.0664	1.9761	-

Table 3.6: order of convergence in the x direction of dispersive equation

N Δt	0.000001	0.000002	0.000004	0.000008
21	-	-	-	-
41	2.0040	2.0040	2.0039	2.0038
81	2.0113	2.0113	2.0111	2.0109
161	-	-	-	-

These three tables show the scheme is second order in both time and space.

Next we conduct large time step test, with the same initial data and exact solution, we notice the speed of the traveling wave is $2\pi - (2\pi)^3$, so one period is $1/(2\pi - (2\pi)^3)$. We use large time step $\Delta t = 1$ compute up to final time $t = 10^{20}/(2\pi - (2\pi)^3)$. We display the numerical solution and the exact solution in Fig.3.3, which shows the wave is resolved very well.

3.3 Linear KS equation

We consider the linear equation

$$u_t = P(\partial_x)u. \tag{3.6}$$

This corresponds to the case

$$P(\partial_x) = -\partial_{xx} - \partial_{xxxx},$$

which is the combination of second and fourth order derivatives.

Fourier analysis shows the problem is well-posed in the sense that

$$||u(\cdot,t)||_{L^2} \leq ||u_0||_{L^2} e^{\frac{t}{4}}, \forall t > 0.$$

According to Case 3 in Theorem 2.1.1 the numerical scheme will be

$$u^{n+1} = (1 + \mathcal{L}^{-1}\Delta t P(\partial_x))u^n e^{\frac{\Delta t}{4}},$$

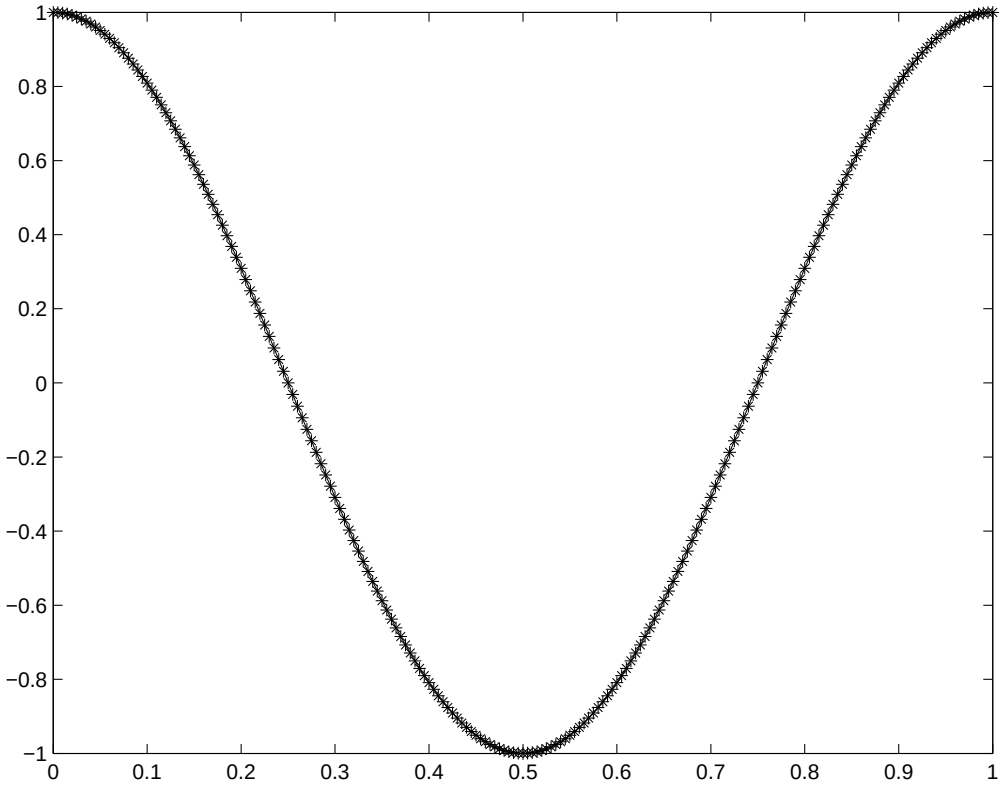

Figure 3.3: large time step test for dispersive equation

where

$$\pounds(\epsilon, \partial_x) = I - \epsilon(-\partial_{xx} - \partial_{xxxx} - 1/4I),$$

with $\frac{\Delta t}{2} \leq \epsilon \leq \Delta t$.

We use central difference to approximate second and fourth order derivative and apply the periodic boundary condition.

$$(D_+D_-)^2 u_j^n = u_{j+2}^n - 4u_{j+1}^n + 6u_j^n - 4u_{j-1}^n + u_{j-2}^n,$$
$$(D_+D_-)u_j^n = u_{j+1}^n - 2u_j^n + u_{j-1}^n.$$

The corresponding matrix for periodic boundary condition and fourth order derivative is

$$\mathbf{X1} = \begin{pmatrix} 6 & -4 & 1 & \dots & 0 & 1 & -4 & 0 \\ -4 & 6 & -4 & 1 & \dots & 0 & 1 & 0 \\ 1 & -4 & 6 & -4 & 1 & \dots & 0 & 0 \\ 0 & 1 & -4 & 6 & -4 & 1 & 0 & \dots \\ & & \ddots & \ddots & \ddots & \ddots & & \\ 0 & 1 & \dots & 0 & 1 & -4 & 6 & -4 \\ 0 & -4 & 1 & \dots & 0 & 1 & -4 & 6 \end{pmatrix},$$

and for the second order derivative is

$$\mathbf{X2} = \begin{pmatrix} -2 & 1 & 0 & \dots & 0 & 0 & 1 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 & \dots \\ & & \ddots & \ddots & \ddots & \ddots & & \\ 0 & 0 & \dots & 0 & 0 & 1 & -2 & 1 \\ 0 & 1 & 0 & \dots & 0 & 0 & 1 & -2 \end{pmatrix}.$$

First we perform a convergence test for our scheme, we choose the initial data to be

$$u(x,0) = cos(2\pi x),$$

and the exact solution is

$$u(x,t) = exp(-16\pi^4 t + 4\pi^2 t)cos(2\pi x).$$

The L_2 error and order of convergence which is computed by formulae (3.2-3.4) are displayed in the following tables.

From this numerical test we see the scheme have numerical convergence order 1 in time and order 2 in space.

The combination of second order and fourth order derivative term in the KS equation, describe a balance between long-wave(small frequency ξ) instability and short wave(large frequency ξ) stability. We notice the frequency $\xi = \sqrt{2}/2$ is an optimal one at which the wave

Table 3.7: L_2 Errors to compute Equation 3.6 by scheme 3.3

N Δt	0.000001	0.000002	0.000004	0.000008
51	0.775E-08	0.8E-08	0.849E-08	0.933
101	0.207E-08	0.23E-08	0.276E-08	0.364
151	0.069E-08	0.092E-08	0.137E-08	0.225
201	0.041E-08	0.06E-08	0.103E-08	0.191

Table 3.8: order of convergence in the t direction $(r1)_{kl}$

N Δt	0.000001	0.000002	0.000004	0.000008
51	-	0.93	0.81	-
101	-	0.97	0.92	-
151	-	0.99	0.94	-
201	-	1.25	1.05	-

grow fastest. Next we perform a numerical test to see how the frequency ξ and time step effect the error between numerical result and the exact solution. We take the initial value to be

$$u(x,0) = e^{\xi ix},$$

and the exact solution is

$$u(x,t) = e^{-\xi^4 t + \xi^2 t} e^{\xi ix}.$$

We take the ξ to be $\frac{\sqrt{2}}{2}, 1, 2\pi, 50\pi, 100\pi$ separately, and for some mode we perform both large time step test and small step test. For large time test we choose Δt =0.0001,0.0003,0.0009, 0.0027,0.0081,0.0243,0.0729,0.2187 and for small step test we use $\frac{1}{1000}$ of the Δt for large time test. And the final time we take for large time test is 5, and 0.005 for small time step test, the numerical result is displayed in Fig.3.4-3.6. From the result we see, there is no big difference as we change θ and ξ when $\xi \leq 1$, whereas the error growth as θ getting closer to 0.5 when $\xi > 1$.

Table 3.9: order of convergence in the x direction $(r2)_{k,l}$

N Δt	0.000001	0.000002	0.000004	0.000008
51	-	-	-	-
101	2.04	2.04	2.04	2.04
151	2.37	2.13	2.01	2.04
201	-	-	-	-

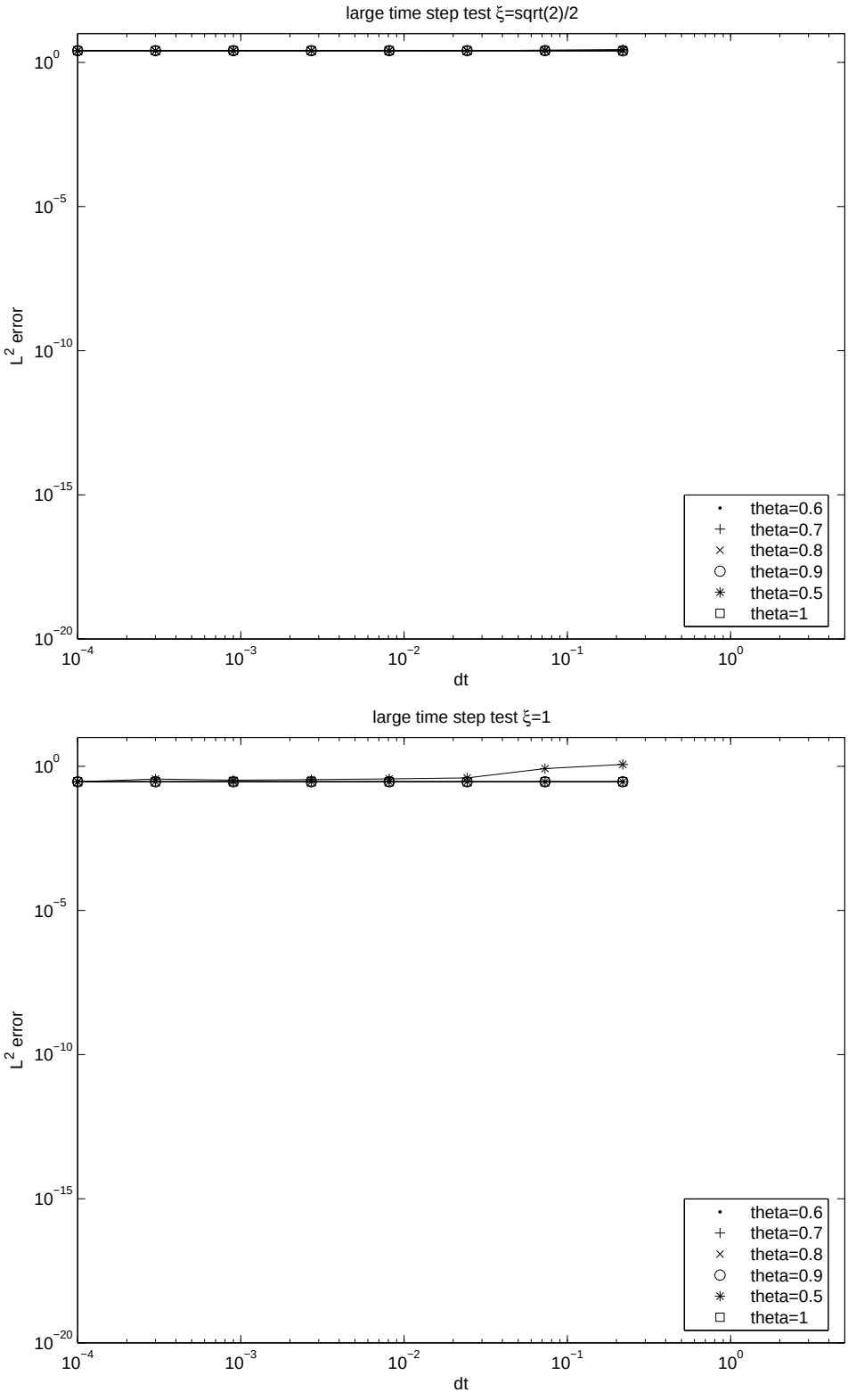

Figure 3.4: large and small time step error comparison for linear KS equation.

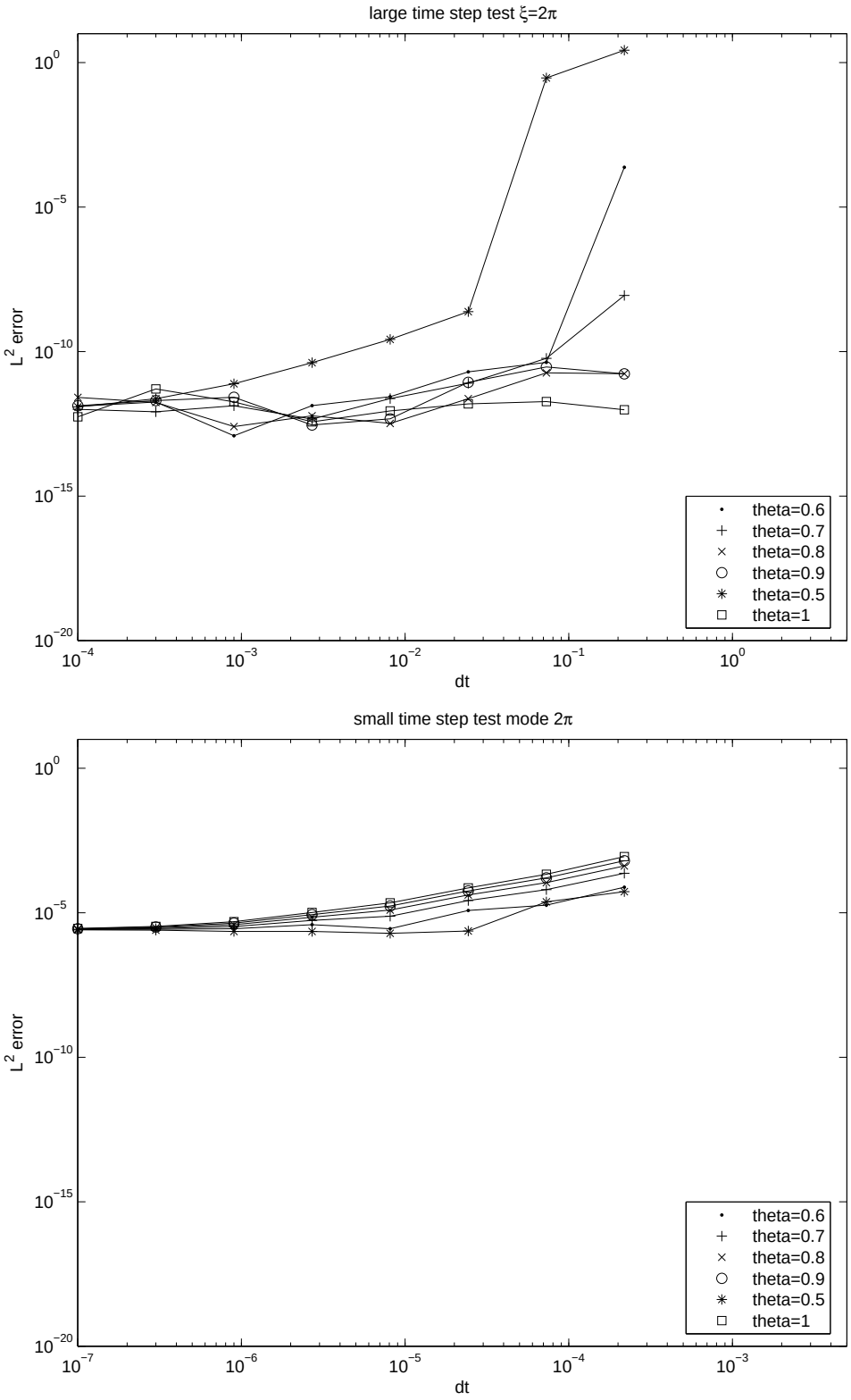

Figure 3.5: large and small time step error comparison for linear KS equation.

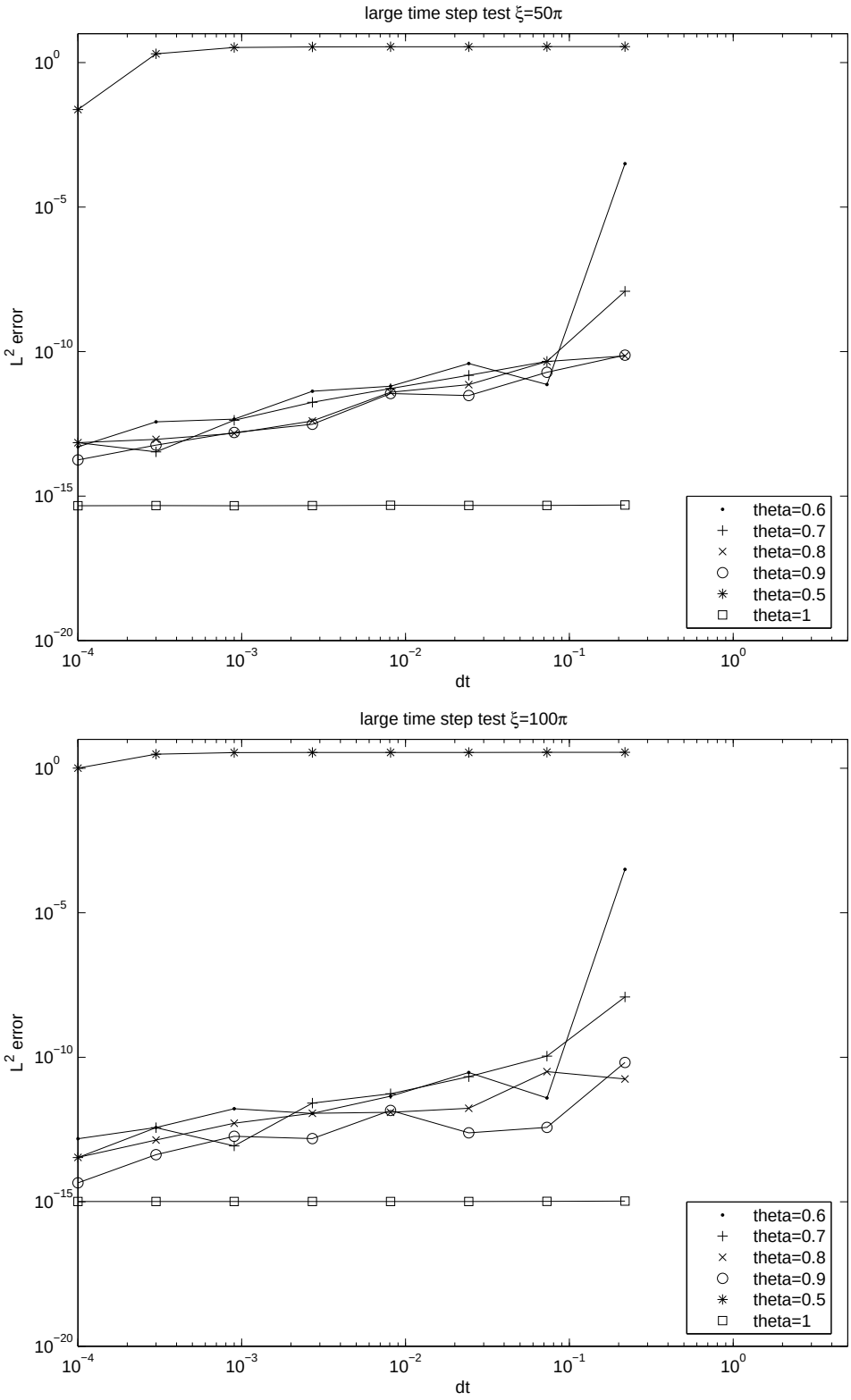

Figure 3.6: large and small time step error comparison for linear KS equation.

CHAPTER 4. The KdV equation

4.1 Background

In this chapter we consider the generalized Korteweg-de Vries(KdV) equation of the form

$$u_t + f(u)_x + u_{xxx} = 0, \tag{4.1}$$

Here $f(u)$ is the given flux function, say $f = u^2/2$. This equation describes the evolution of long one-dimensional waves in many physical settings, in which the dispersion u_{xxx} counterbalances the nonlinear term $f(u)_x$. The KdV equation is well-known to satisfy infinitely many conservation laws, the most fundamental properties are the preservation of the mass and energy

$$\int_{\mathbb{R}} u(x,t) = \int_{\mathbb{R}} u_0(x)dx \quad (mass), \tag{4.2}$$

$$\int_{\mathbb{R}} u^2(x,t)dx = \int_{\mathbb{R}} u_0^2(x)dx \quad (energy). \tag{4.3}$$

where $u_0(x)$ is a given initial data decaying at far field. Due to the higher order spatial derivatives in (4.1) any explicit time discretization would restrict the time step to order of $(\Delta x)^3$. This is not suitable for long time simulation. On the other hand it is believed that the numerical method preserving both mass and energy is more stable and suitable for long-time integration.

Our goal here is to design a second order finite difference method for (4.1): (i) to preserve both the mass and energy relations (4.2) and (4.3) respectively, at the discrete level; and (ii) to be stable for large time step.

We note that, in a recent work [10], a finite volume method was introduced in order to preserve both mass and energy; but in their method two equations for u and u^2 are evolved.

Our method is more efficient since only one equation is used. For other numerical methods developed for the KdV equation ranging from finite difference method, finite element methods to spectral methods, see e.g. [2],[3],[16],[35].

4.2 Strang splitting and time discretization

In (4.1) we take $f(u) = (u^2/2)$. The initial value problem (4.1) becomes

$$u_t + uu_x + u_{xxx} = 0, u(x,0) = u_0(x). \tag{4.4}$$

First we describe the operator splitting strategy, see [17]. The strategy is alternately to solve the equation

$$u_t + uu_x = 0, u(x,0) = u_0(x), \tag{4.5}$$

and the linear dispersive equation

$$u_t + u_{xxx} = 0, u(x,0) = u_0(x). \tag{4.6}$$

Let S_t be the solution operator associated with the conservation law (4.5), we write the unique weak solution to (4.5) as $u(x,t) = S_t u_0(x)$. Similarly we denote the solution operator associated with the linear dispersive equation (4.6) by A_t. Then we approximate the solution of (4.4) by

$$u_{\Delta t}(x, n\Delta t) = [A_{\Delta t} \circ S_{\Delta t}]^n u_0(x). \tag{4.7}$$

When implementing this approach, both A_t and S_t must be replaced by numerical methods. In time direction we use Crank-Nicolson

$$\frac{u^{n+1} - u^n}{\Delta t} + u^{n+1/2} u_x^{n+1/2} + u_{xxx}^{n+1/2} = 0.$$

In practice, for each time step evolution we use the Strang splitting method

$$u^{n+1} = [S_{\frac{\Delta t}{2}} \circ A_{\Delta t} \circ S_{\frac{\Delta t}{2}}] u^n,$$

to obtain second order accuracy in time direction.

4.3 Spacial discretization and Energy preserving scheme

First we define

$$D_0 u_j = \frac{u_{j+1} - u_{j-1}}{2h},$$
$$D_+ u_j = \frac{u_{j+1} - u_j}{h},$$
$$D_- u_j = \frac{u_j - j_{j-1}}{h}.$$

We propose a numerical scheme

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + D_0 D_+ D_-(u_j^{n+1/2}) = 0, \tag{4.8}$$
$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{1}{6h}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}) = 0. \tag{4.9}$$

In (4.8),4.9 we use $D_0 D_+ D_-$ to approximate u_{xxx}, and to approximate uu_x we apply the semi-discrete scheme

$$\frac{d}{dt}u_n + \frac{1}{3}(u_{n+1} + u_n + u_{n-1})(\frac{u_{j+1} - u_{j-1}}{2h}) = 0,$$

which is refered to as 1/3 dispersive scheme, and has been studied by Levermore and Liu [24]. Then we have the following result.

Theorem 4.3.1. *Assume $\sum_j (u_j^n)^2 < \infty$ and $\Sigma_j u_j^0 < \infty$ for $j \in \mathbb{Z}$, then scheme (4.8) and (4.9) satisfies $\sum_j (u_j^{n+1})^2 = \sum_j (u_j^n)^2$, and $\sum_j (u_j^{n+1}) = \sum_j (u_j^n)$ if we assume the numerical solution has compact support.*

Proof. Multiplying both sides of (4.8) by the factor $u_j^{n+1/2} = \frac{u_j^{n+1} + u_j^n}{2}$, we obtain

$$(u_j^{n+1})^2 - (u_j^n)^2 + 2\Delta t u_j^{n+1/2} D_0 D_+ D_- u_j^{n+1/2} = 0.$$

Summing this over the values of j, $j \in \mathbb{Z}$, and omitting the subscript on the norm $||.||_2$ we obtain

$$||u^{n+1}||^2 - ||u^n||^2 = -2\Delta t \Sigma_{j\in\mathbb{Z}} u_j^{n+1/2} D_0 D_+ D_- u_j^{n+1/2}. \tag{4.10}$$

Next we prove the right hand side of (4.10) is 0. For simplicity we omit the superscript for $u_j^{n+1/2}$.

$$\begin{aligned}
&\Sigma_{j\in\mathbb{Z}} u_j D_0(D_+D_- u_j) \\
&= \Sigma_{j\in\mathbb{Z}}(u_j D_+D_- u_{j+1} - u_j D_+D_- u_{j-1}) \\
&= \Sigma_{j\in\mathbb{Z}}(u_j(u_{j+2} - 2u_{j+1} + u_j) - u_j(u_j - 2u_{j-1} + u_{j-2}) \qquad (4.11)\\
&= \Sigma_{j\in\mathbb{Z}}(u_j u_{j+2} - u_{j-2}u_j) + (u_{j+1}u_j - u_j u_{j-1}) \qquad (4.12)\\
&= 0, \qquad (4.13)
\end{aligned}$$

where we have assumed $u_j = 0$ for $|j|$ large. So, (4.8) conserves the L_2 norm at discrete level.

Next we multiply both sides of (4.9) by $u_j^{n+1/2} = \frac{u_j^{n+1}+u_j^n}{2}$ and summing this over the values of j to obtain

$$||u^{n+1}||^2 - ||u^n||^2 = -2\Delta t/(3h)\Sigma_{j=1}^J u_j^{n+1/2}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}). \quad (4.14)$$

We still omit the superscript to prove the right hand side of (4.14) is 0.

$$\begin{aligned}
&\Sigma_{j\in\mathbb{Z}} u_j(u_{j+1} + u_j + u_{j-1})(u_{j+1} - u_{j-1}) \\
&= \Sigma_{j\in\mathbb{Z}}(u_j(u_{j+1})^2 - u_j(u_{j-1})^2 + (u_j)^2 u_{j+1} - (u_j)^2 u_{j-1}) \qquad (4.15)\\
&= 0. \qquad (4.16)
\end{aligned}$$

This completes the proof of L_2 norm conservation $\sum_j (u_j^{n+1})^2 = \sum_j (u_j^n)^2$.

We can prove the conservation of mass $\sum_j u_j^{n+1} = \sum_j u_j^n$ in the same way.

4.4 Numerical implementation

First we make a N partition of the computational domain $[a, b]$, so $\Delta x = \frac{b-a}{\Delta x}$.

In order to implement (4.9) each time step we need an internal iteration. We rewrite (4.9) as

$$\frac{u_j^{n+1} + u_j^n}{2} - u_j^n + \frac{\Delta t}{12h}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}) = 0,$$

which can be written as

$$u_j^{n+1/2} + \frac{\Delta t}{12h}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}) - u_j^n = 0, \tag{4.17}$$

where

$$u_j^{n+1/2} = \frac{1}{2}(u_j^{n+1} + u_j^n).$$

Then (4.17) becomes a system of nonlinear equations when j goes from 1 to N. For simplicity we use V_j to denote $u_j^{n+1/2}$, and $\lambda = \frac{\Delta t}{12h}$, the system becomes

$$\begin{aligned} V_1 + \lambda(V_2^2 - V_{N-1}^2 + V_1(V_2 - V_{N-1})) - u_1 &= 0, \\ V_j + \lambda(V_{j+1}^2 - V_{j-1}^2 + V_j(V_{j+1} - V_{j-1})) - u_j &= 0, j = 2, 3, \cdots, n-1, \\ V_N + \lambda(V_2^2 - V_{N-1}^2 + V_N(V_2 - V_{N-1})) - u_N &= 0. \end{aligned}$$

In order to solve this system of nonlinear equations $F(V) = 0$, where $V = (v_1, \cdots, v_N)$. We apply the Newton's method

$$\begin{aligned} J(V_k)s_k &= -F(V_k), \\ V_{k+1} &= V_k + s_k, \end{aligned}$$

where J(V) is the Jacobian (matrix) of F(V).

The iterations are conducted until the following criterion is satisfied:

$$|F(V_k)| < \delta, \tag{4.18}$$

where δ is a pre-given tolerance number.

4.5 Numerical experiments

Example 4.5.1. Consider the one-soliton solution of equation (4.1)with $f(u) = \frac{u^2}{2}$ and

$$u_0(x,t) = Asech^2(kx - wt - x_0), \tag{4.19}$$

where $A = 12k^2$,$w = 4k^3$ We take $k = 0.3$ and $x_0 = 0$, and take $u(x, 0)$ as the initial condition.

Table 4.1: Example 5.1, numerical errors and convergence rate of u at t=0.02

N	L^2	order	L^∞	order	L^1	order
50	5.02E-05	-	2.55E-05	-	1.53E-04	-
100	1.31E-05	1.93	6.96E-06	1.9	4.03E-05	1.92
150	3.32E-06	1.98	1.8E-06	1.95	1.02E-05	1.98
200	8.5E-07	1.97	4.73E-07	1.93	2.72E-06	1.91

Table 4.2: Example 5.1, numerical errors and convergence rate of u^2 at t=0.02

N	L^2	order	L^∞	order	L^1	order
50	7.9E-05	-	5.21E-05	-	1.52E-04	-
100	2.1E-05	1.91	1.4E-05	1.90	4.03E-05	1.92
150	5.33E-06	1.98	3.61E-06	1.96	1.02E-05	1.98
200	1.34E-06	1.99	9.56E-07	1.91	2.72E-06	1.91

Firstly we check the accuracy and convergence rates of the method. The solution region is taken to be (-10,10). We assign the boundary conditions with the exact solution to make the numerical simulation accurate. We take small $\Delta t = 0.0001$, and perform the computation on the grid of 50,100,150,200 cells, we fix the final time at $t = 0.02$. The L^1, L^2, L^∞ errors and the corresponding convergence rates are presented in Table 5.1 and Table 5.2. We can see from the tables that our method is second-order accurate for both u and u^2.

Secondly, we use this solution to test the stability and robustness of the method in long-time integration. We take the computational domain to be (-10,10), with periodic boundary conditions. We notice that the soliton has an advection speed of 0.36, so a period of the soliton is $\frac{500}{9} \approx 55.5$. We thus use this observation to make comparison between the numerical solution and the exact solution. We take 400 cells $\Delta t = 0.01$, and plot the waves at time 55.5 (1 period), 111 (2 periods) and 166.5 (3 periods) 222 (4 periods), the solutions are displayed in Fig.4.1, we see this method is very stable in long-time simulation. We notice that the phase error becomes severe at longer time, because the numerical error accumulates as the computational time becomes longer, we can reduce the phase error by making finer grid cell or taking smaller time step Δt.

Example 4.5.2. We consider the two-soliton solution of (5.1) with the same $f(u)$

$$u(x,t) = 12\Big(k_1^2 exp(\theta_1) + k_2^2 \exp(\theta_2) + 2(k_1-k_2)^2 \exp(\theta_1+\theta_2) + a^2(k_2^2\exp(\theta_1)) + k_1^2 exp(\theta_2)\exp(\theta_1+\theta_2)\Big) \Big/ \Big(1+\exp\theta_1+\exp\theta_2+a^2\exp\theta_1+\theta_2\Big),$$

where

$$k_1 = 0.4, k_2 = 0.6 \quad , \quad a^2 = (\frac{(k_1-k_2)2}{(k_1+k_2)})^2 = \frac{1}{25}, \tag{4.20}$$

$$\theta_1 = k_1 x - k_1^3 t + x_1 \quad , \quad \theta_2 = k_2 x - k_2^3 t + x_2, \tag{4.21}$$

$$x_1 = 4 \quad , \quad x_2 = 15 \tag{4.22}$$

We set up the computational domain to be (-40,40), and apply the periodic boundary condition. Initially we have two solitons the tall one is to the left of the short one. Both solitons travel from left to right and speed of the tall one is bigger than the speed of the short one. So after some time the tall one catches up the short one. After they goes out from the right end, they reappears from left due to the periodic boundary condition. Then the tall one will catch up the short one again, this process will keep going on. To show the stability and robustness of our scheme in long-time simulation and to display the interactions of solitons, we display the (x,t)-contour of the solution in the time interval (0,500), see Fig.4.2. In this picture we see the numerical solution preserve a stable and smooth shape for the solitons

Next we compare the numerical result with the exact solution. The computational domain is still taken to be (-40,40), and $\Delta t = 0.1$, we plot both the numerical solution and exact solution at $t = 20, 40, 120$ in Fig.4.3. The solid line represent the exact solution and the * line is the numerical solution, we see they agree each other very well. This example has been used in [9].

Example 4.5.3. Zabusky-Kruskal's problem [39] Consider the KdV equation

$$u_t + uu_x + (0.022^2)u_{xxx} = 0, \tag{4.23}$$

with periodic boundary condition. The solution is described by Zabusky and Kruskal in 1965. $t = T_B = 1/\pi$ is the breakdown time, and at $t = 3.6T_B$ 8 solutions is generated. Using scheme

(4.8), (4.9), we perform computations for $\Delta t = 0.01/\pi$, $\Delta x = \frac{2}{400}$, and we stop at the critical time $t = 3.6/\pi$. These three waves displayed in Fig.4.4 agree with the result given by Zabusky and Kruskal in 1965 see [39].

Next we compare our scheme with the leap frog scheme used by Zabusky and Kruskal

$$u_j^{n+1} = u_j^{n-1} - \frac{\Delta t}{3\Delta x}(u_{j+1}^n + u_j^n + u_{j-1}^n)(u_{j+1}^n - u_{j-1}^n) - \frac{0.022^2 \Delta t}{\Delta x^3}(u_{i+2}^j - 2u_{i+1}^j + 2u_{i-1}^j - u_{i-2}^j). \tag{4.24}$$

The numerical result by (4.24) with small time step $\Delta t = \frac{5*10^{-5}}{\pi}, \Delta x = \frac{2}{400}$ is displayed in [39]. The waveform will blow up for $\pi t > 21$. By comparison we perform the computation by our scheme (4.8), (4.9) until the final time $t = 21/\pi$, we display the numerical result at time $t = 19.9/\pi$ and $t = 21/\pi$ in Fig.4.5. The numerical result at $t = 19.9$ agree with the waveform given by the twelve-points scheme see [39], and at the break down time $21/\pi$ the waveform is still smooth by our scheme. And we notice the time step we use is $\Delta t = 0.01/\pi$ while in twelve-point scheme $\Delta t = 0.005/\pi$ and in leap frog scheme $\Delta t = 0.00005/\pi$, our time step is much larger than the other two schemes.

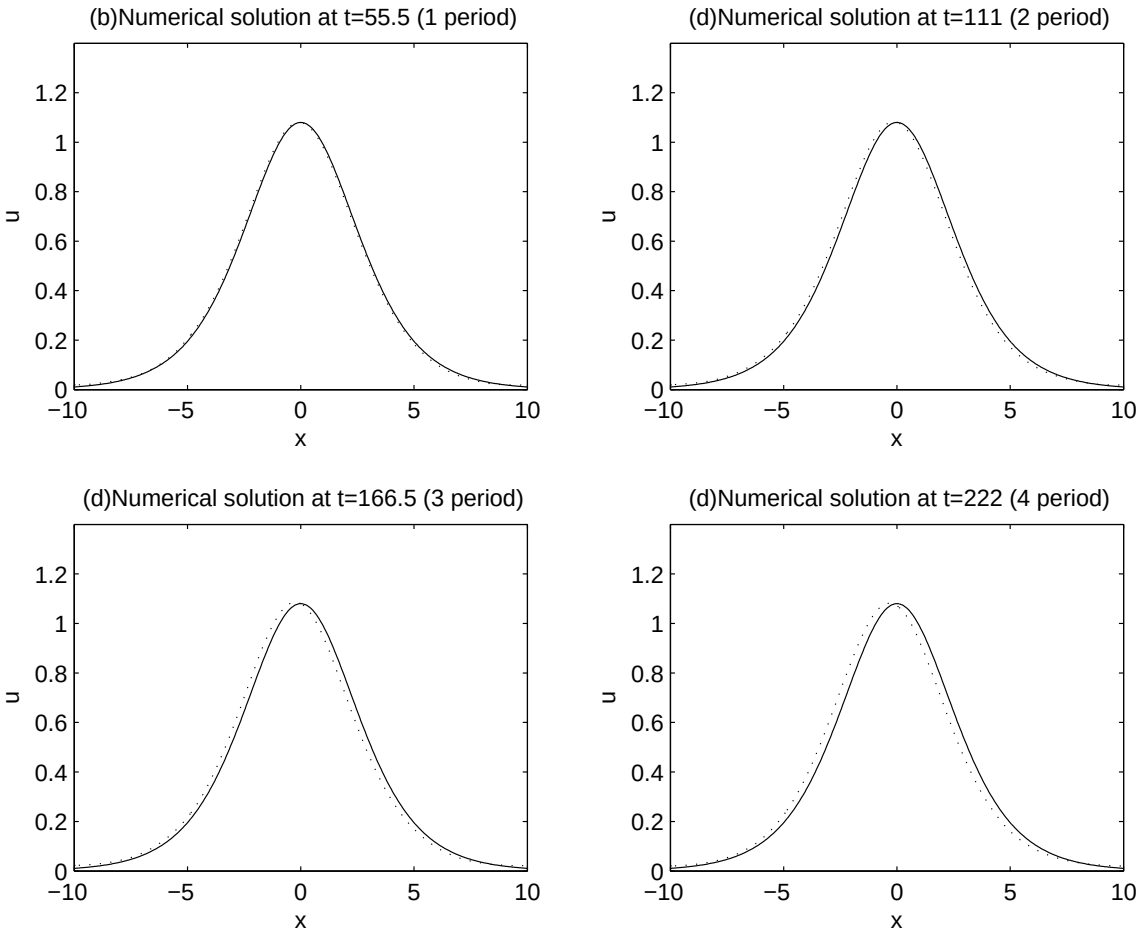

Figure 4.1: Example 4.5.1, solution plots at different times.

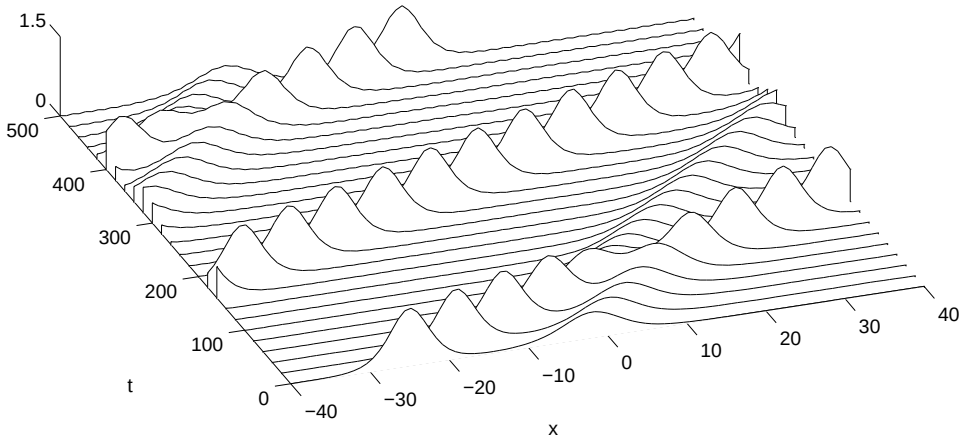

Figure 4.2: Example 4.5.2, (x,t)-contour of the solution in (0,500).

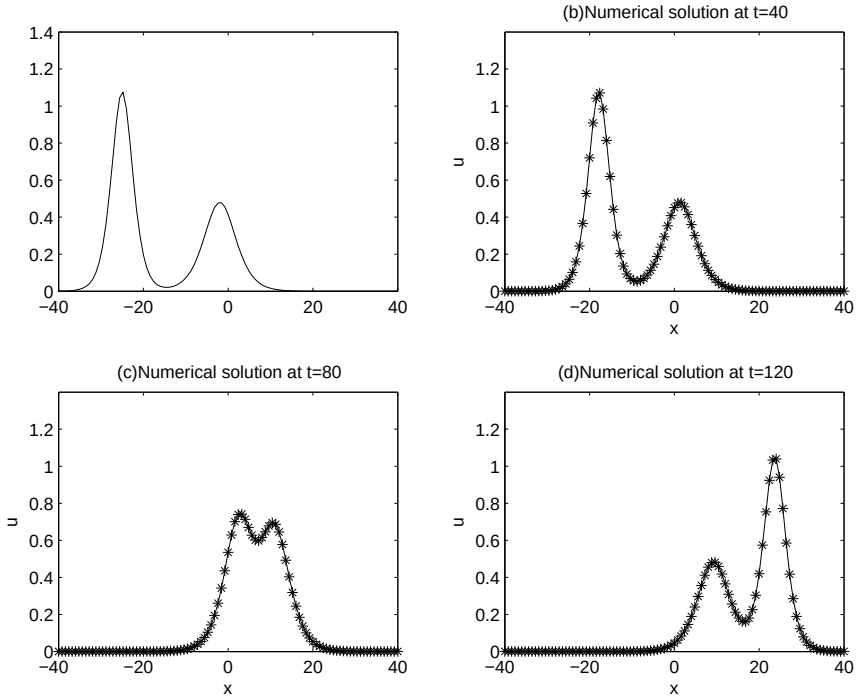

Figure 4.3: Example 4.5.2, solution plots at different times.

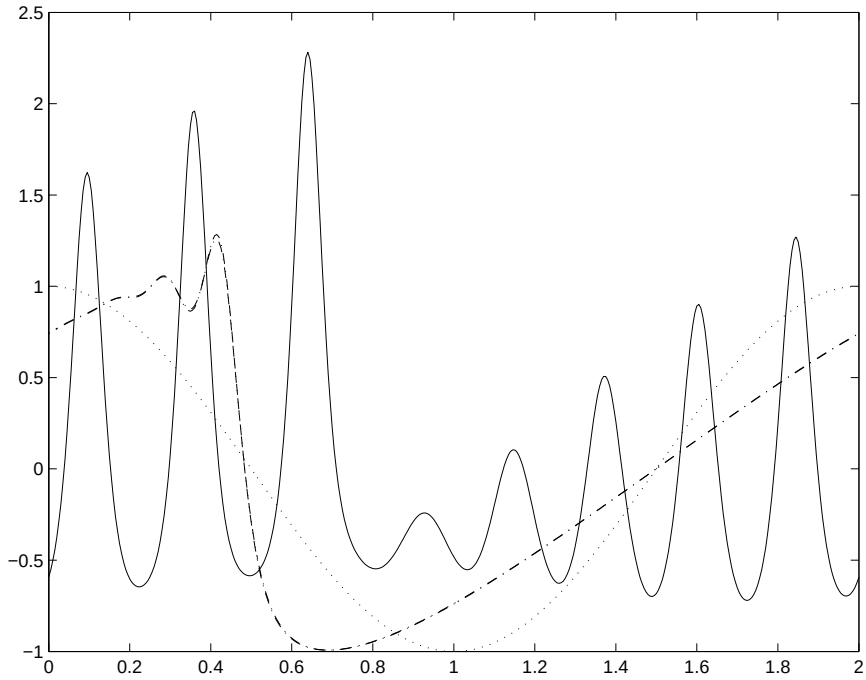

Figure 4.4: Waves for Zabusky-Kruskal's problem.

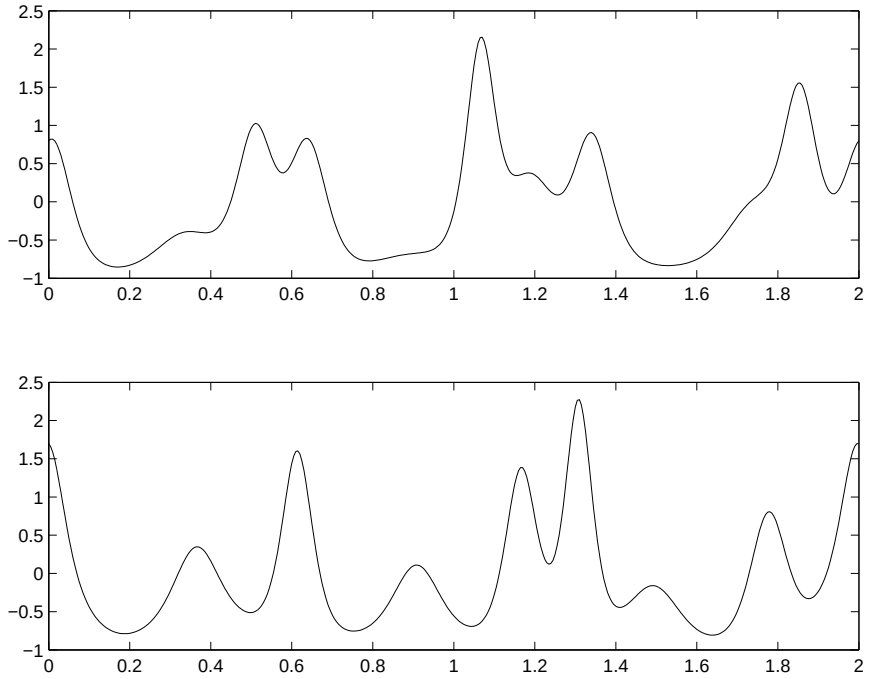

Figure 4.5: Waves for Zabusky-Kruskal's problem.

CHAPTER 5. The Swift-Hohenberg equation

In this chapter we consider a class of diffusion equations of the form

$$u_t = \alpha u - (1+\Delta)^2 u + f(u), x \in D \subseteq R^2, \tag{5.1}$$

where

$$f(u) = \frac{dF(u)}{du}.$$

This equation has been extensively studied in literatures ,see e.g.[8], [7].

One example is the Swift-Hohenberg equation

$$u_t = \alpha u - (1+\partial_{xx})^2 u - u^3,$$

where α is a bifurcation parameter and the potential function

$$F(u) = \frac{1}{2}(\alpha - 1)u^2 - \frac{u^4}{4},$$

then

$$f(u) = \frac{dF(u)}{du} = (\alpha - 1)u - u^3.$$

The Swift-Hohenberg equation is first proposed in 1977 by Swift and Hohenberg [31] and later has been applied to the problem of laser [23] and Taylor-Couette flow [15]. The swift-Hohenberg equation is interesting in pattern formation; with quadratic and cubic nonlinearities, it exhibits many stable equilibrium states. The pattern of the equilibrium states depends on both the parameters and the spatial domain, see [38],[5],[27],[29],[28].

5.1 Algorithm equation as design and energy analysis

We rewrite 5.1 as

$$u_t = Lu + \frac{dF}{du}, \tag{5.2}$$

where

$$L = -\Delta^2 - 2\Delta = -\partial_{xxxx} - \partial_{yyyy} - 2\partial_{xx} - 2\partial_{yy} - 2\partial_{xxyy}.$$

Let $u(x,t)$ be a smooth solution of (5.2), then subject to the Dirichlet boundary condition

$$u = \Delta u = 0, (x,y) \in \partial\Omega. \tag{5.3}$$

The equation (5.2) admits a nonincreasing energy functional

$$\Psi = \int_\Omega \{\frac{1}{2}(\Delta u)^2 - (\nabla u)^2 - F(u)\}dxdy, \tag{5.4}$$

$$\frac{d\Psi}{dt} = -\int_\Omega \Big(\frac{\partial u}{\partial t}\Big)^2 dxdy < 0. \tag{5.5}$$

We expect the numerical scheme preserve (5.5) at discrete level. We discretize (5.2) in time to obtain

$$\frac{u^{n+1} - u^n}{\Delta t} = L\Big(\theta u^{n+1} + (1-\theta)u^n\Big) + \frac{F(u^{n+1}) - F(u^n)}{u^{n+1} - u^n}, \tag{5.6}$$

where $u^n = u(t_n, x, y), t_n = n\Delta t$, θ is a parameter in $(0, 1]$. The use of parameter θ servers to smoothen the numerical solution $u^n(x)$. Second term of right hand side is the divided difference of $\frac{dF}{du}$in (5.6). We now show that for certain θ, the semi-discrete scheme (5.6) is stable. We summarize the result in the following:

Theorem 5.1.1. *Consider the semi-discrete scheme (5.6) with any given potential function F(u).*

1).If $\theta = \frac{1}{2}$, the scheme (5.6) is stable in the sense of

$$\Psi[u^{n+1}] = \Psi[u^n] - \int_\Omega \frac{(u^{n+1} - u^n)^2}{\Delta t} dxdy \le \Psi[u^n] \tag{5.7}$$

for arbitrary Δt.

2). If $\theta > \frac{1}{2}$, then the scheme (5.6) is stable in the sense of

$$\Psi[u^{n+1}] \le \Psi[u^n]$$

for all $\Delta t \leq \frac{1}{2\theta-1}$.

Proof. 1) (5.6) can be rewritten as

$$\frac{u^{n+1}-u^n}{\Delta t}+\big(\partial_{xxxx}+2\partial_{xx}+\partial_{yyyy}+2\partial_{yy}+2\partial_{xxyy}\big)(\theta u^{n+1}+(1-\theta)u^n)-\frac{F(u^{n+1})-F(u^n)}{u^{n+1}-u^n}=0. \tag{5.8}$$

We multiply $u^{n+1}-u^n$ on both sides and integrate over Ω, using integration by parts, together with the admissible boundary condition $u=u_{xx}=0, u=u_{yy}=0$, to obtain

$$\begin{aligned}
&\int_\Omega \Big\{\frac{(u^{n+1}-u^n)^2}{\Delta t}\\
&-2(\theta u_x^{n+1}+(1-\theta)u_x^n)(u_x^{n+1}-u_x^n)+(\theta u_{xx}^{n+1}+(1-\theta)u_{xx}^n)(u_{xx}^{n+1}-u_{xx}^n)\\
&-2(\theta u_y^{n+1}+(1-\theta)u_y^n)(u_y^{n+1}-u_y^n)+(\theta u_{yy}^{n+1}+(1-\theta)u_{yy}^n)(u_{yy}^{n+1}-u_{yy}^n)\\
&+2(\theta u_{xy}^{n+1}+(1-\theta)u_{xy}^n)(u_{xy}^{n+1}-u_{xy}^n)-(F(u^{n+1})-F(u^n))\Big\}dxdy=0.
\end{aligned}$$

We regroup the terms to obtain

$$\begin{aligned}
&\int_\Omega \{\frac{(u^{n+1}-u^n)^2}{\Delta t}\\
&-((u_x^{n+1})^2-(u_x^n)^2)+\frac{1}{2}((u_{xx}^{n+1})^2-(u_{xx}^n)^2)+(\theta-\frac{1}{2})\Big((u_{xx}^{n+1}-u_{xx}^n)^2-2(u_x^{n+1}-u_x^n)^2\Big)\\
&-((u_y^{n+1})^2-(u_y^n)^2)+\frac{1}{2}((u_{yy}^{n+1})^2-(u_{yy}^n)^2)+(\theta-\frac{1}{2})\Big((u_{yy}^{n+1}-u_{yy}^n)^2-2(u_y^{n+1}-u_y^n)^2\Big)\\
&+\Big((u_{xy}^{n+1})^2-(u_{xy}^n)^2\Big)+2(\theta-\frac{1}{2})(u_{xy}^{n+1}-u_{xy}^n)^2-[F(u^{n+1})-F(u^n)]\}dxdy=0.
\end{aligned} \tag{5.9}$$

By the definition of the energy functional Ψ we have

$$\begin{aligned}
&\Psi[u^{n+1}]-\Psi[u^n]+\int_\Omega\Big\{\frac{(u^{n+1}-u^n)^2}{\Delta t}\\
&+(\theta-\frac{1}{2})\Big((u_{xx}^{n+1}-u_{xx}^n)^2-2(u_x^{n+1}-u_x^n)^2\Big)\\
&+(\theta-\frac{1}{2})\Big((u_{yy}^{n+1}-u_{yy}^n)^2-2(u_y^{n+1}-u_y^n)^2\Big)\\
&+(\theta-\frac{1}{2})\Big((u_{xy}^{n+1}-u_{xy}^n)^2\Big\}dxdy=0.
\end{aligned} \tag{5.10}$$

We see when $\theta = 1/2$ the scheme satisfy the energy identity (5.7) at the discrete level.

2) From (5.10), we have

$$\begin{aligned}
&\Psi[u^{n+1}] - \Psi[u^n] + \int_\Omega \Big\{ \frac{(u^{n+1}-u^n)^2}{\Delta t} \\
&+ (\theta - \frac{1}{2})\Big((u^{n+1}_{xx} - u^n_{xx})^2 + 2(u^{n+1}_{xx} - u^n_{xx})(u^{n+1} - u^n)\Big) \\
&+ (\theta - \frac{1}{2})\Big((u^{n+1}_{yy} - u^n_{yy})^2 + 2(u^{n+1}_{yy} - u^n_{yy})(u^{n+1} - u^n)\Big) \\
&+ (\theta - \frac{1}{2})(u^{n+1}_{xy} - u^n_{xy})\Big\} dxdy = 0.
\end{aligned}$$

We add and substract $2(\theta - \frac{1}{2})(u^{n+1} - u^n)^2$ to obtain

$$\begin{aligned}
&\Psi[u^{n+1}] - \Psi[u^n] + \int_\Omega \Big\{ (\frac{1}{\Delta t} - 2\theta + 1)(u^{n+1} - u^n)^2 \\
&+ (\theta - \frac{1}{2})\Big((u^{n+1}_{xx} - u^n_{xx}) + (u^{n+1} - u^n)\Big)^2 \\
&+ (\theta - \frac{1}{2})\Big((u^{n+1}_{yy} - u^n_{yy}) + (u^{n+1} - u^n)\Big)^2 \\
&+ (\theta - \frac{1}{2})(u^{n+1}_{xy} - u^n_{xy})^2\Big\} dxdy.
\end{aligned}$$

We see when $\theta > 1/2$

$$\Psi[u^{n+1}] - \Psi[u^n] \le (2\theta - 1 - \frac{1}{\Delta t}) \int_\Omega (u^{n+1} - u^n)^2 dxdy.$$

Then we can conclude that when $\Delta t \le \frac{1}{2\theta - 1}$, the energy functional Ψ is nonincreasing i.e.

$$\Psi[u^{n+1}] \le \Psi[u^n].$$

The proof is now complete. □

5.2 Numerical Algorithms

We consider the Swift-Hohenberg equation

$$u_t = \alpha u - (1 + \partial_x^2)^2 u - u^3, 0 < x < L, t > 0. \tag{5.11}$$

We take $\theta = 1/2$ as an example. To implement (5.6) in each time step we need to apply an internal iteration and discretization in space.

1).Internal iteration

$$[1+\frac{\Delta t}{2}(1-\alpha)]u_{n,k+1}+\frac{1}{2}\Delta t\partial_{xxxx}u_{n,k+1}+\frac{\Delta t}{4}(u_{n,k}^2+u_{n,k}u^n+u_n^2)u_{n,k+1}=$$
$$u_n-\frac{\Delta t}{2}((1-\alpha)+\partial_{xxxx}+2\partial_{xx})u_n-\frac{\Delta t}{4}u_n^3-\Delta t\partial_{xx}u_{n,k}, \tag{5.12}$$

where $u_{n,k+1}, u_{n,k}$denote the numerical solution at k+1 and k iteration step.

2).Spatial discretization

To discretize ∂_{xx} and ∂_{xxxx} we use central difference such that

$$\begin{aligned}\partial_{xxxx}u_i^n &\approx \frac{u_{i+2}^n-4u_{i+1}^n+6u_i^n-4u_{i-1}^n+u_{i-2}^n}{\Delta x^4},\\ \partial_{xx}u^i &\approx \frac{u_{i+1}^n-2u_i^n+u_{i-1}^n}{(\Delta x)^2}.\end{aligned}$$

5.3 Numerical test

We still consider (5.11) with $\alpha=0.3$ and $u=u_{xx}=0$ at $x=0, x=L$.

We use the initial data

$$u_0=Asin(\frac{\pi x}{L}),\quad A=\frac{1}{10}.$$

Since no exact solution is available, we compare the numerical solution u_j^n with a reference solution u_j^{*n}, which is obtained by the same scheme taking smaller $\Delta t=0.0001$. The L_2 and L_1 error is evaluated by

$$||e||_{L^2}=\sum_j|u_j-u_j^*|^2\Delta x,$$
$$||e||_{L^1}=\sum_j|u_j-u_j^*|\Delta x.$$

1) Accuracy test. We perform this test for the optimal choice $\theta=\frac{1}{2}$, and the initial function we choose is

$$u_0=Asin(\frac{\pi x}{L}), A=\frac{1}{10},$$

and we take $\alpha=0.3$. We take small $\Delta t=0.0001$ and large grid cell number $N=1600$, and use the corresponding numerical solution as a reference solution. From the table we see this

Table 5.1: order of convergence of scheme (5.12)

N	h	L^2error	order	L^∞error	order	L^1error	order
100	0.03	1.808E-03	-	1.476E-03	-	2.82E-03	-
200	0.015	8.44E-04	1	6.892E-04	0.9999	1.316E-03	1
400	0.0075	3.617E-04	1	2.953E-04	1	5.64E-04	1
800	0.00375	1.205E-04	-	9.845E-05	-	1.88E-04	-

test we see the scheme is first order in space, which shows the numerical convergence of the scheme.

The second test we do is to show how the change of θ effects the stability of the scheme. We fix $N = 100$, $tmax = 40$ (From numerical test we see that the numerical solution reach steady state solution from this time). We change θ and Δt then compare the solution with the exact solution in L_2 norm. We consider $\theta = \frac{1}{2}, \Delta t = 0.01$ as the reference solution.

Table 5.2: stability test for scheme (5.12)

	$\Delta t = 0.01$	$\Delta t = 0.1$	$\Delta t = 1$
$\theta = \frac{1}{6}$	unstable	unstable	unstable
$\theta = \frac{1}{3}$	unstable	unstable	8.18E-03
$\theta = \frac{2}{5}$	unstable	unstable	9.177E-03
$\theta = \frac{1}{2}$	0	4.922E-04	8.329E-03
$\theta = \frac{2}{3}$	7.485E-05	7.734E-04	9.679E-03
$\theta = \frac{5}{6}$	1.549E-04	1.313E-03	1.151E-02
$\theta = 1$	2.406E-04	1.594E-03	1.512E-02

This numerical test is consistent with the energy analysis that when $\theta \geq \frac{1}{2}$ the numerical scheme is stable.

5.4 Pattern formation

In the following numerical experiment we will show how the parameters and the spatial domain effect the equilibrium states. We consider the $1 - D$ Swift-Hohenberg equation in the form of

$$\partial_t u = \alpha u - (\partial_{xx} + a)^2 u + \beta u^2 - u^3, x \in [0, L] \tag{5.13}$$

where α, a, β and L are parameters.

We slightly modify (5.12) to obtain the numerical scheme for (5.13) as following

$$\frac{u^{n+1}-u^n}{\Delta t} = (-\partial_{xxxx} + \alpha - a^2)\frac{u^n_{k+1}+u^n}{2} - 2a\partial_{xx}\frac{u^n_k+u^n}{2}$$
$$+\frac{1}{3}\beta(u^n_k+u^n)u^n_{k+1} + \frac{1}{3}\beta(u^n)^2 - \frac{1}{4}((u^n_k)^2 + u^n_k u^n + (u^n)^2)u^n_{k+1} - \frac{1}{4}(u^n)^3.$$

We still use the same initial function

$$u_0 = Asin(\frac{\pi x}{L}), A = \frac{1}{10}.$$

Then we have the following numerical result for one-dimensional Swift-Hohenberg equation pattern formation.

Case 1:First we fix $a > 0$, $\beta > 0$ and $L > 0$ and change α. From Fig.5.1 we see when $\alpha \leq 0$ a trivial solution appears. For $\alpha > 0$ the situation is different. When $0 < \alpha < 1.5$ a stable sine-like pattern appears and as α increases the period becomes less. The numerical test tells us for $0 < \alpha < 0.55$, 3.5 periods appear in $[0, 20]$ and for $0.55 < \alpha < 1.5$, 2.5 periods appear. When $\alpha > 1.5$, the pattern changes as shown in Fig.5.2.

Case 2: Keep other parameter values constant: e.g. $\alpha = 1, \beta = 0, L = 20$ and vary smoothly the constant a. See Fig.5.3 and Fig.5.4. As a grows, sine-like periodic pattern is observed, which is symmetric and the period becomes smaller as a increases.

Case 3: In this case we take random data as the initial value to perform numerical tests. We fix $\alpha = 1, a = 2$, and vary coefficient β to see how the pattern changes, See Fig.5.5 and Fig.5.6.

1.When $0 < \beta < 6.155$, still sine-like pattern appears, but the symmetric property breaks as β increases.

2.When $6.155 < \beta < 6.17$, there is no stable pattern for β in this region. For example when $\beta = 6.158$, we observe four different patterns among tests performed starting for random initial data.

3. When $\beta > 6.18$, the pattern retains stable shape, just the amplitude increases as β increases.

Finally, we fix all the parameters in equation (5.13) to be $\alpha = 2, \beta = 0, a = 2$ to see how the pattern changes when we increase the computational domain L. From Fig5.7, we see when L is small enough, the solution tends to be trivial solution; When $L = 6$ the final state is nontrivial and it does change sign on the interval $(0, L)$; When L become bigger, the solution tends to trivial solution again, after $L = 9$ the final profile has the stable sine-like pattern, the number of periods increases as L grows.

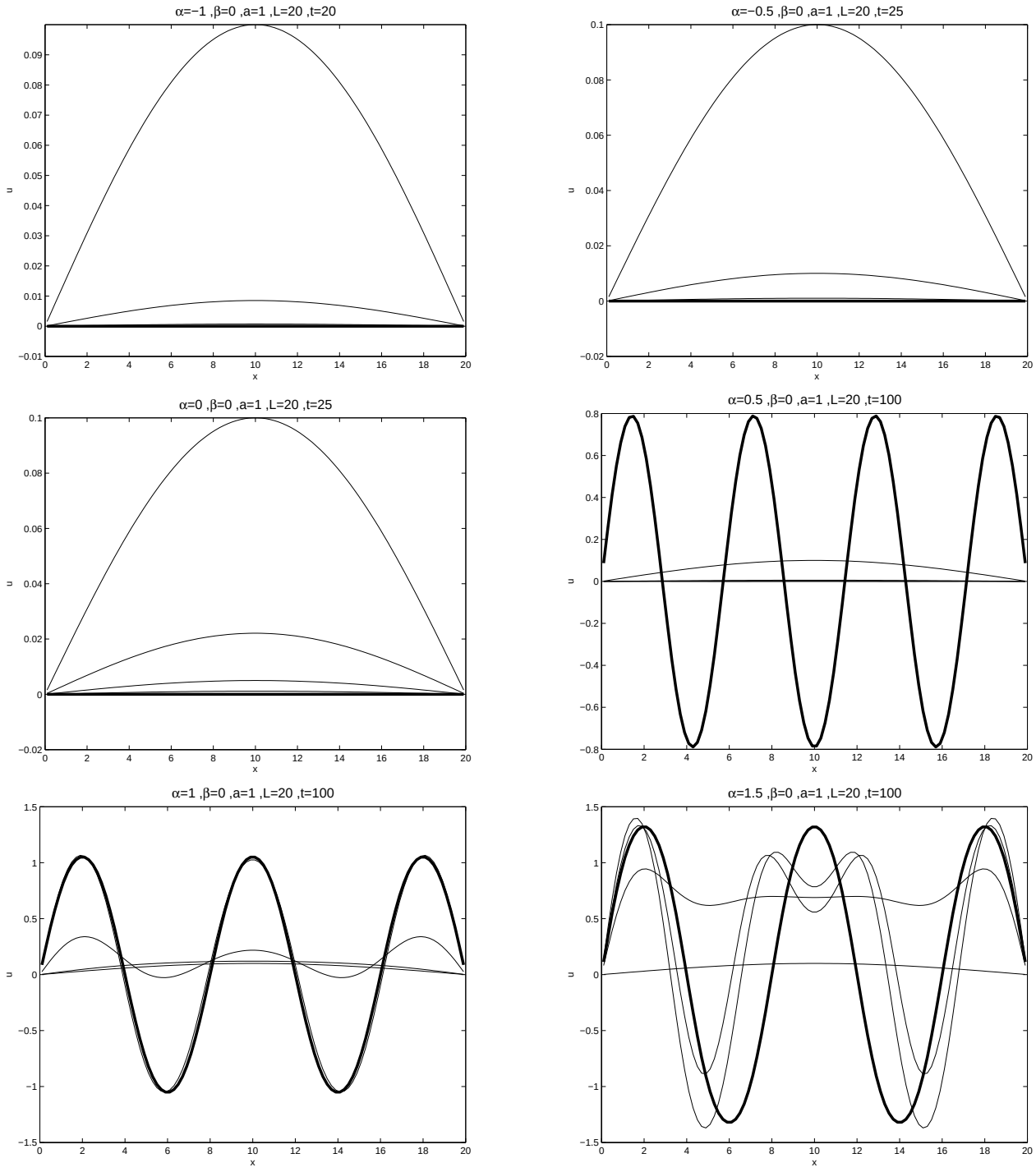

Figure 5.1: Swift-Hohenberg equation pattern formation 1.

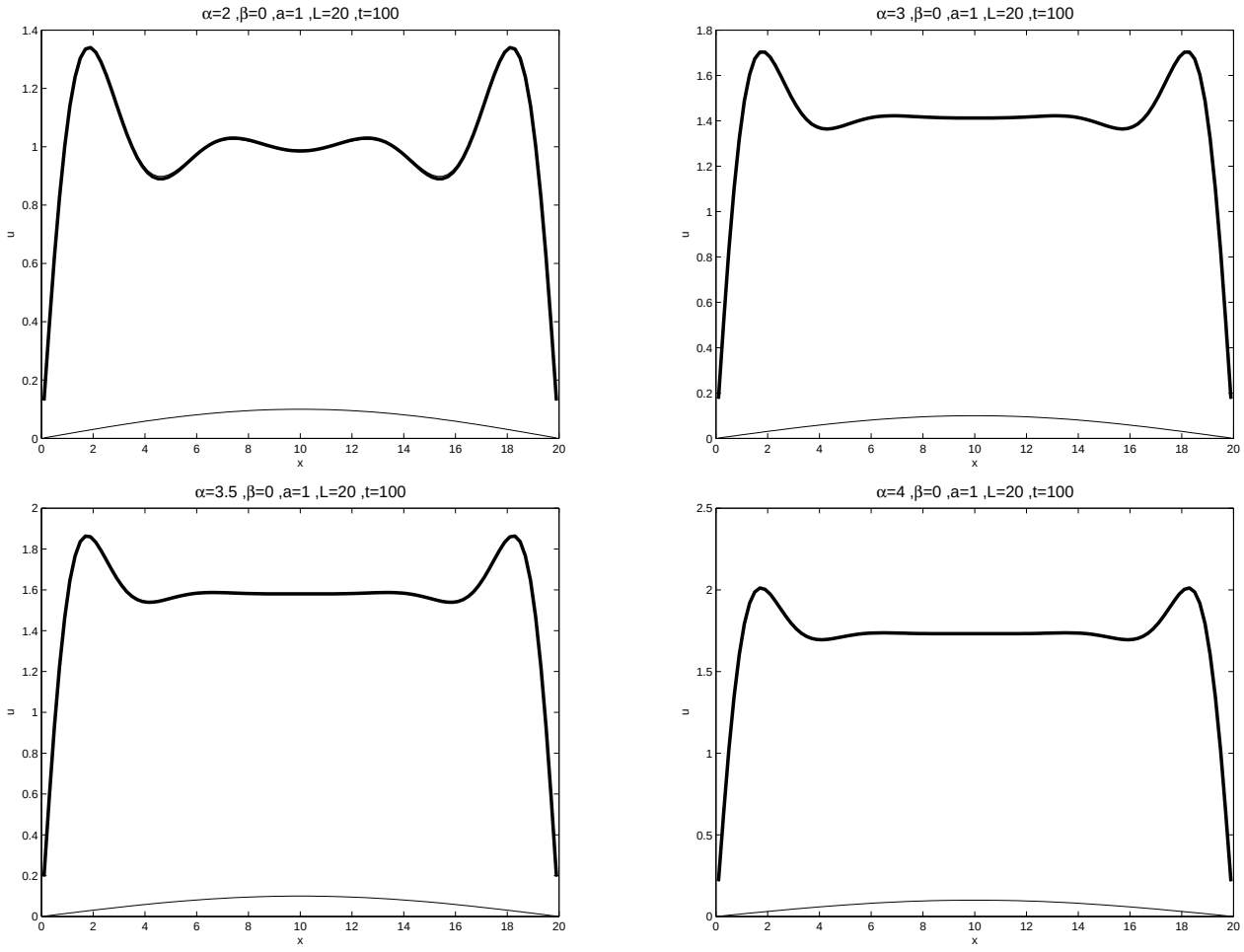

Figure 5.2: Swift-Hohenberg equation pattern formation 2.

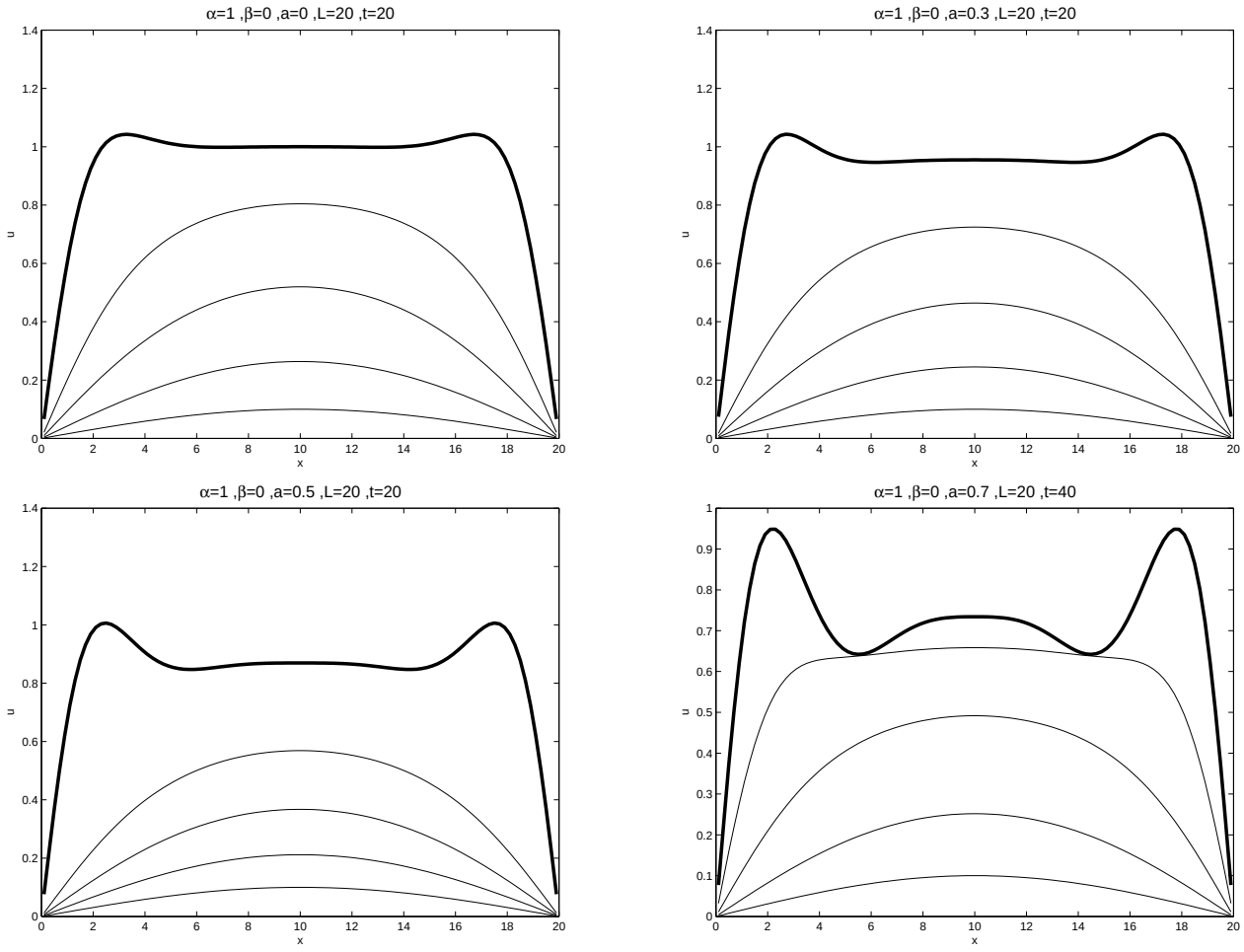

Figure 5.3: Swift-Hohenberg equation pattern formation 3.

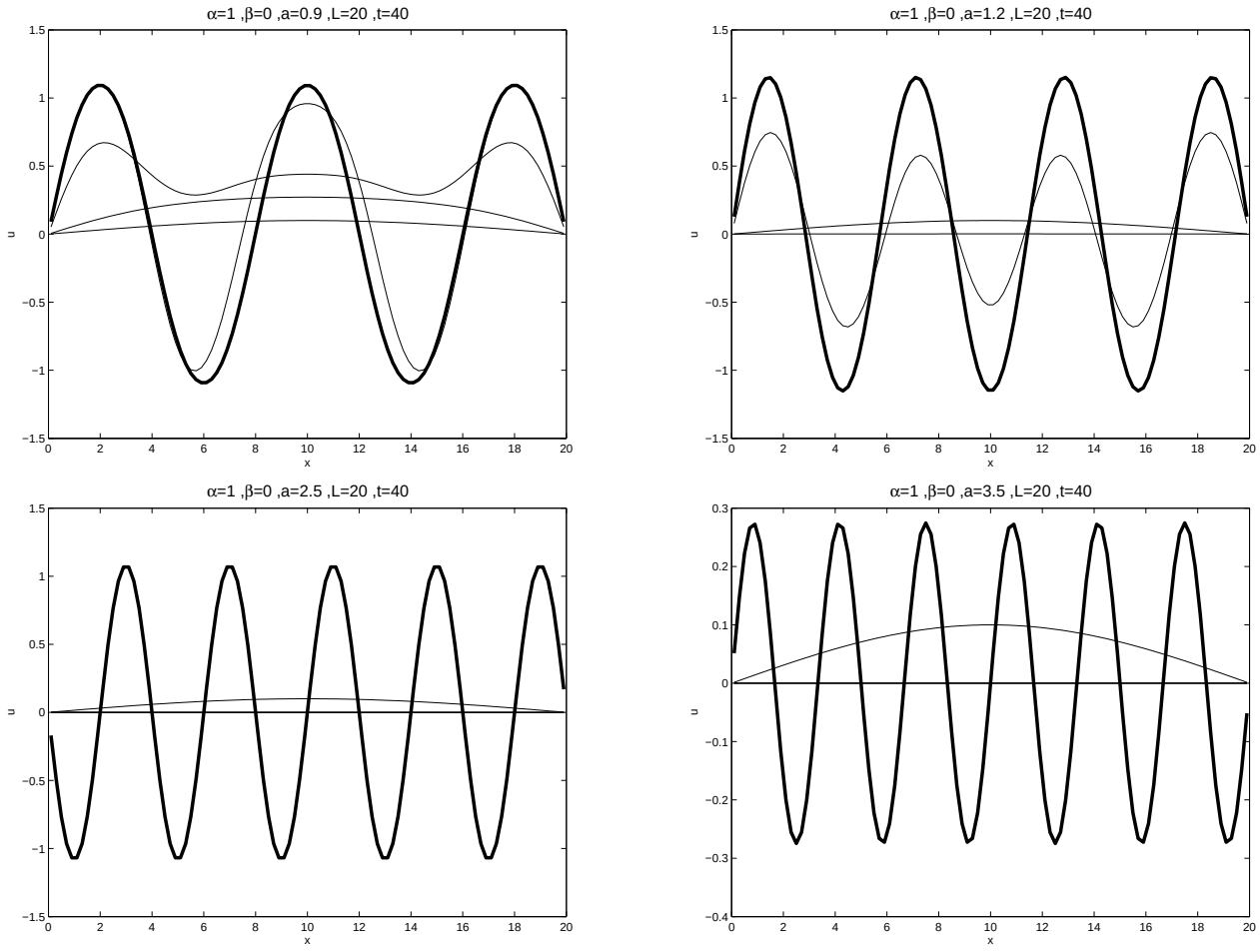

Figure 5.4: Swift-Hohenberg equation pattern formation 4.

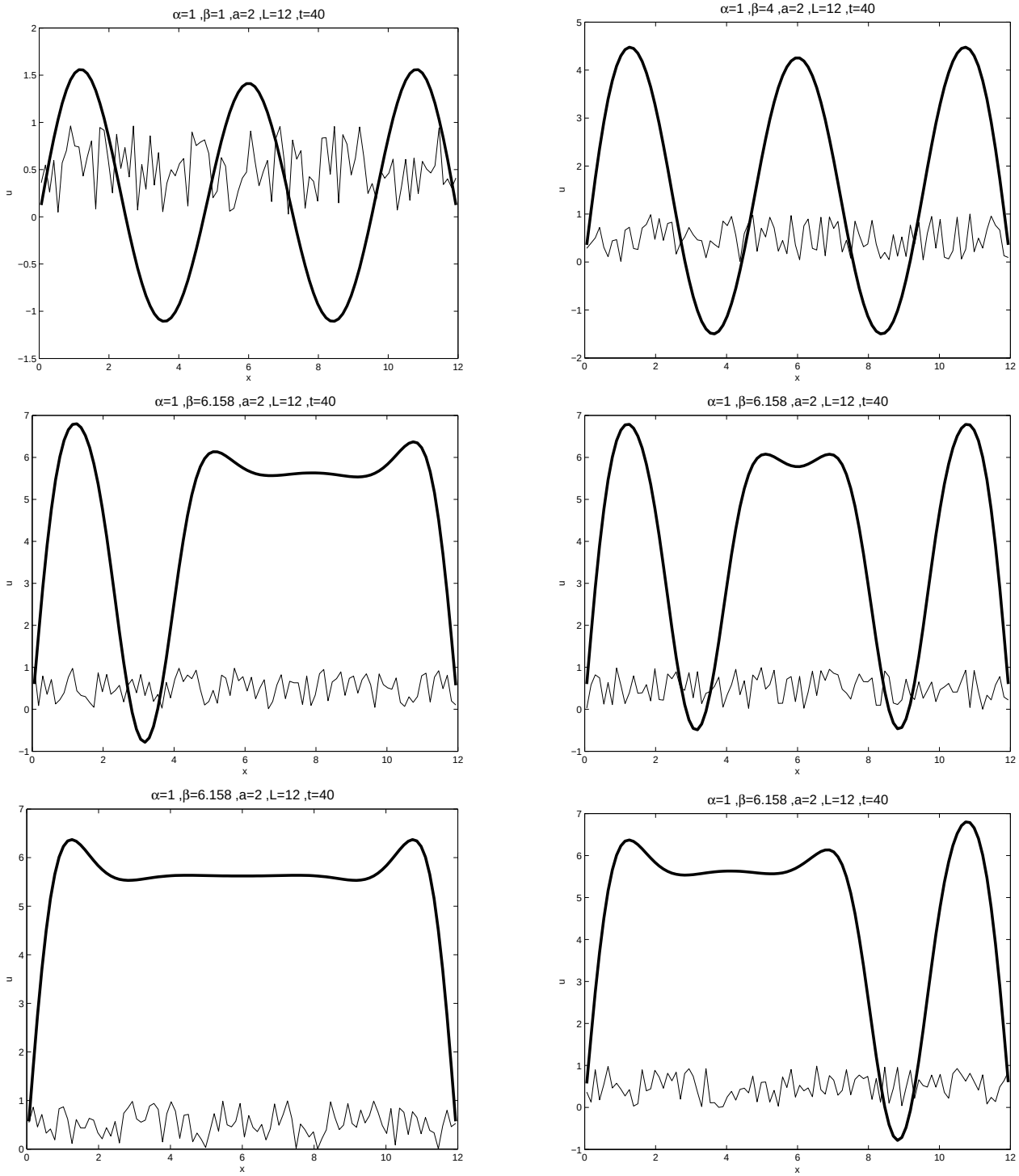

Figure 5.5: Swift-Hohenberg equation pattern formation 5.

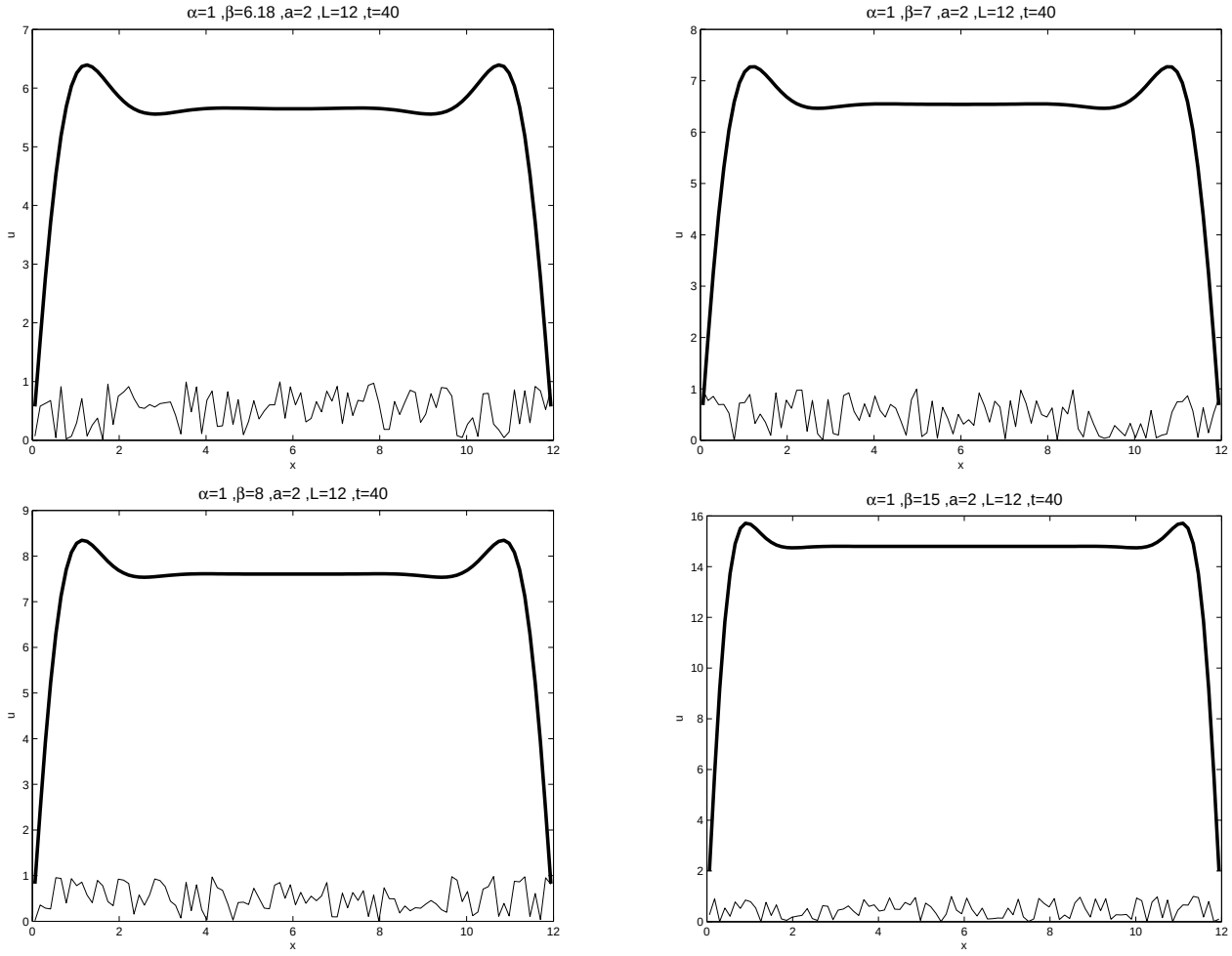

Figure 5.6: Swift-Hohenberg equation pattern formation 6.

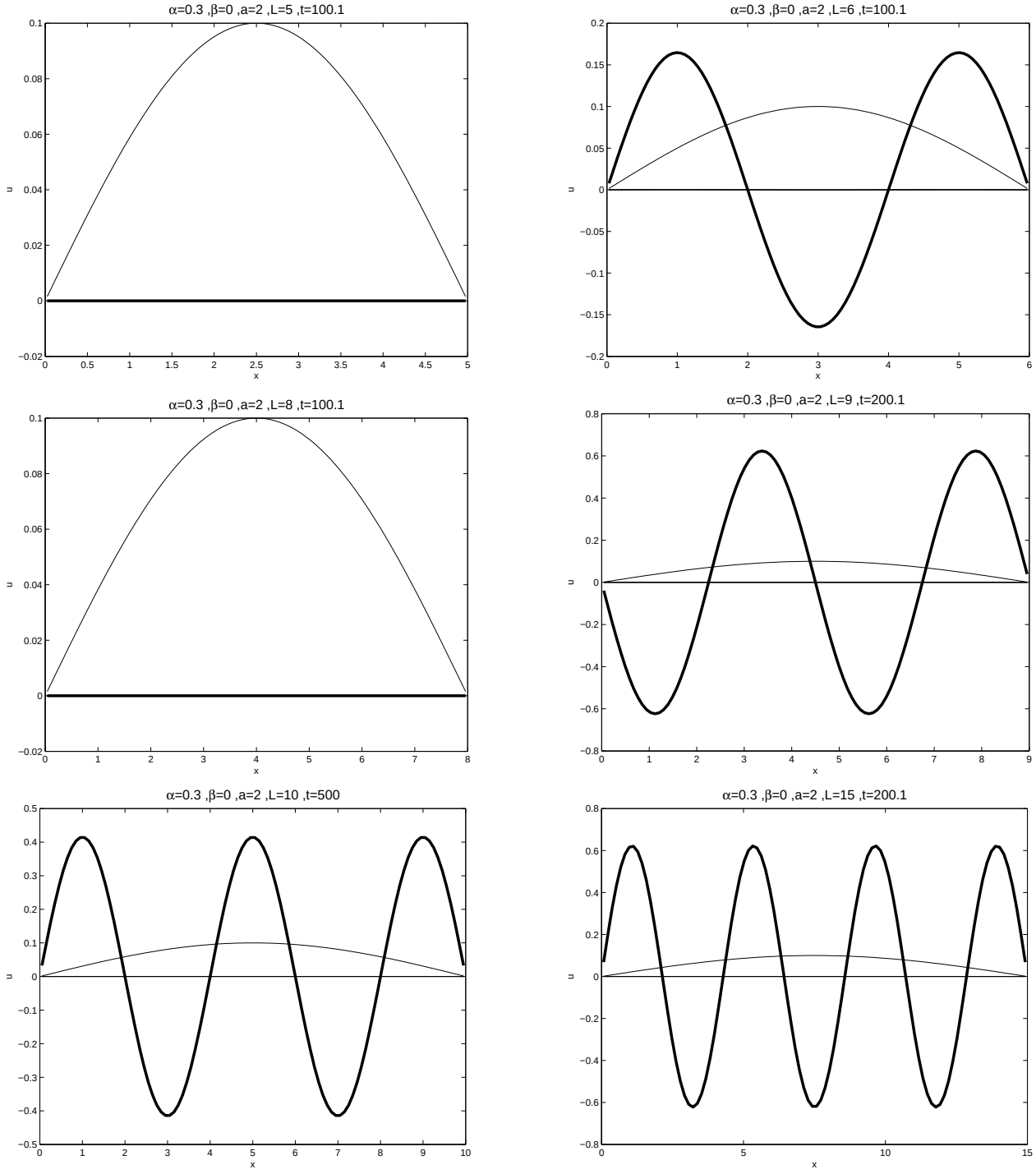

Figure 5.7: Swift-Hohenberg equation pattern formation 7.

CHAPTER 6. The Kuramoto-Sivashinsky type equation

6.1 Background

The Kuramoto-Sivashinsky equation

$$u_t + uu_x + au_{xx} + bu_{xxxx} = 0, \tag{6.1}$$

has attracted considerable attention over the last decades. The well-poseness of equation (6.1) is proved in [32]. When the coefficients a and b are both positive, its second order term has a destabilizing effect and the fourth order term has a stabilizing effect, with the nonlinear term providing a mechanism for energy transfer from low to high wavenumbers. It arises in various contexts and admits different types of solutions like traveling wave solution, see [18],[19],[25],[37]. From a dynamical system point of view, it is an interesting PDE that can exhibit chaotic solutions.

6.2 Numerical scheme for generalized Kuramoto-Sivashinsky equation

We consider the general form of the Kuramoto-Sivashinsky type equation

$$u_t + uu_x + (a(u)u_x)_x + (b'(u)(b(u)_x)_x)_x + (c(u)u_{xx})_{xx} = 0, \quad x \in \mathbb{R}, \tag{6.2}$$

with an initial condition

$$u(x,0) = u_0(x), x \in \mathbb{R}, \tag{6.3}$$

where a,b,c are continuous functions of u. We consider the problem on [0,1] with periodic boundary condition, we multiply both sides of (6.2) by u and integrate by parts over [0,1] to obtain the identity

$$\frac{1}{2}\frac{d}{dt}\int_0^1 u^2(x,t)dx - \int_0^1 a(u)(u_x)^2dx + \int_0^1 c(u)(u_{xx})^2dx = 0. \tag{6.4}$$

Next we will give a lemma to show that the identity (6.4) implies the L_2 stability of solution u of (6.2).

Lemma 6.2.1. *Assume $0 \le a(u) \le M$ and $c(u) \ge r \ge 0$. Let u be a smooth periodic solution to (6.2), then the identity (6.4) implies*

$$||u(\cdot,t)||^2 \le ||u_0||^2 e^{\frac{M^2}{2r}}, \tag{6.5}$$

where $||\cdot||$ denotes the L_2 norm in $L^2(0,1)$.

Proof. From (6.4), with the lower bound of $c(u)$ and upper bound of $a(u)$, we have

$$\frac{1}{2}\frac{d}{dt}||u||^2 = M||u_x||^2 - r||u_{xx}||^2. \tag{6.6}$$

For periodic and smooth u we apply the Cauchy's inequality with ϵ to obtain

$$||u_x||^2 \le \frac{M}{4r}||u||^2 + \frac{r}{M}||u_{xx}||^2.$$

Thus (6.6) yields

$$\frac{1}{2}\frac{d}{dt}||u||^2 \le \frac{M^2}{4r}||u||^2,$$

from which (6.5) follows. □

We propose the following scheme in order to preserve (6.4) at discrete level

$$\begin{aligned}\frac{u_j^{n+1}-u_j^n}{\Delta t} + \frac{1}{6h}(v_{j+1}+v_j+v_{j-1})(v_{j+1}-v_{j-1}) + D_+(a(v_j)D_-v_j)\\ +D_0(b'(v_j)D_+D_-b(v_j)) + D_+D_-(c(v_j)D_+D_-v_j) = 0,\end{aligned} \tag{6.7}$$

where $v = \frac{u^{n+1}+u^n}{2}$. This scheme for $a = 1, b = 0, c = u$ has been studied in ([1]).

Next we will prove the numerical scheme (6.7) indeed keep the identity (6.4) at discrete level which also implies the L_2 stability of scheme (6.7).

Lemma 6.2.2. *Assume $u_j, j = 1, 2, ..., N$ is the solution of equation (6.7), and it is periodical, i.e. $v_{j+N} = v_j$ then scheme (6.7) satisfies*

$$\frac{1}{2}\sum_j \frac{((u_j^{n+1})^2 - (u_j^n)^2)}{\Delta t} = \sum_j a(v_j)(D_+v_j)^2 - \sum_j c(v_j)(D_+D_-v_j)^2. \tag{6.8}$$

Proof. Multiplying both sides of (6.7) by the factor $v_j = \frac{u_j^{n+1}+u_j^n}{2}$, and sum this over the values of j we obtain

$$||u^{n+1}||^2 - ||u^n||^2 = -2\Delta t/(3h)\sum_j v_j(v_{j+1}+v_j+v_{j-1})(v_{j+1}-v_{j-1})$$
$$+\sum_j v_j D_+(a(v_j)D_-v_j) + \sum_j v_j D_0(b'(v_j)D_+D_-b(v_j)) - \sum_j (v_j)D_+D_-(c(v_j)D_+D_-v_j).$$

First we compute

$$\begin{aligned}
&\sum_j v_j(v_{j+1}+v_j+v_{j-1})(v_{j+1}-v_{j-1}) \\
= &\sum_j v_j(v_{j+1})^2 - v_j(v_{j-1})^2 + (v_j)^2v_{j+1} - (v_j)^2v_{j-1} \\
= &\ u_N u_{N+1}^2 - u_0u_1^2 + u_N^2u_{N+1} - u_0^2u_1.
\end{aligned}$$

By applying the periodicity of u_i, we have $u_0 = u_n, u_1 = u_{n+1}$, which give us

$$u_N u_{N+1}^2 - u_0u_1^2 + u_N^2u_{N+1} - u_0^2u_1 = 0.$$

And by summation by parts formula [26] and the periodicity assumption we have

$$\begin{aligned}
\sum_j (v_jD_+(a(v_j)D_-v_j)) &= -\sum_j a(v_j)(D_+v_j)^2, \\
\sum_j v_jD_+D_-c(v_j)D_+D_-v_j &= \sum_j c(v_j)(D_+D_-v_j)^2.
\end{aligned}$$

and we notice

$$\begin{aligned}
&\Sigma_j v_j D_0(b'(v_j)D_+D_-b(v_j)) \\
= &\ \Sigma_j(-D_0v_j)b'(v_j)D_+D_-b(v_j) \\
= &\ \Sigma_j(-D_0b(v_j))D_+D_-b(v_j) \\
= &\ \Sigma_j\frac{1}{2\Delta x^2}(b(v_{j+1})-b(v_{j-1}))(b(v_{j+1})-2b(v_j)+b(v_{j-1})) \\
= &\ 0.
\end{aligned}$$

This completes the proof. □

6.3 Numerical tests

In this section we use Newton iteration to perform several numerical tests for the scheme (6.7).

Example 1:

Accuracy test.

We compute the Kuramoto-Sivashinsky equation

$$u_t + uu_x + u_{xx} + u_{xxxx} = 0, \tag{6.9}$$

with the initial condition

$$u(x,0) = c + \frac{15}{19}\sqrt{\frac{11}{19}}(-9tanh(k(x-x_0)) + 11tanh^3(k(x-x_0))).$$

The exact solution is

$$u(x,t) = c + \frac{15}{19}\sqrt{\frac{11}{19}}(-9tanh(k(x-ct-x_0)) + 11tanh^3(k(x-ct-x_0))).$$

This numerical example has been used in [36]. From (6.7) we have the numerical scheme for (6.9)

$$\begin{aligned}\frac{u_j^{n+1}-u_j^n}{\Delta t} + \frac{1}{6h}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}) \\ +D_+D_-u_j^{n+1/2} + D_+D_-D_+D_-u_j^{n+1/2} = 0.\end{aligned} \tag{6.10}$$

Table 6.1: $C = 5, k = \frac{1}{2}\sqrt{\frac{11}{19}}, x_0 = -12$. Newmann boundary condition. Uniform meshes with N cells at time $t = 1$.

N	L^2	order	L^∞	order
50	2.69	-	1.45	-
100	6.91E-01	1.96	4.50E-01	1.69
200	1.71E-01	2.02	1.13E-01	2.00
400	4.25E-02	2.01	2.79E-02	2.01

We display the numerical result in Fig.6.1.

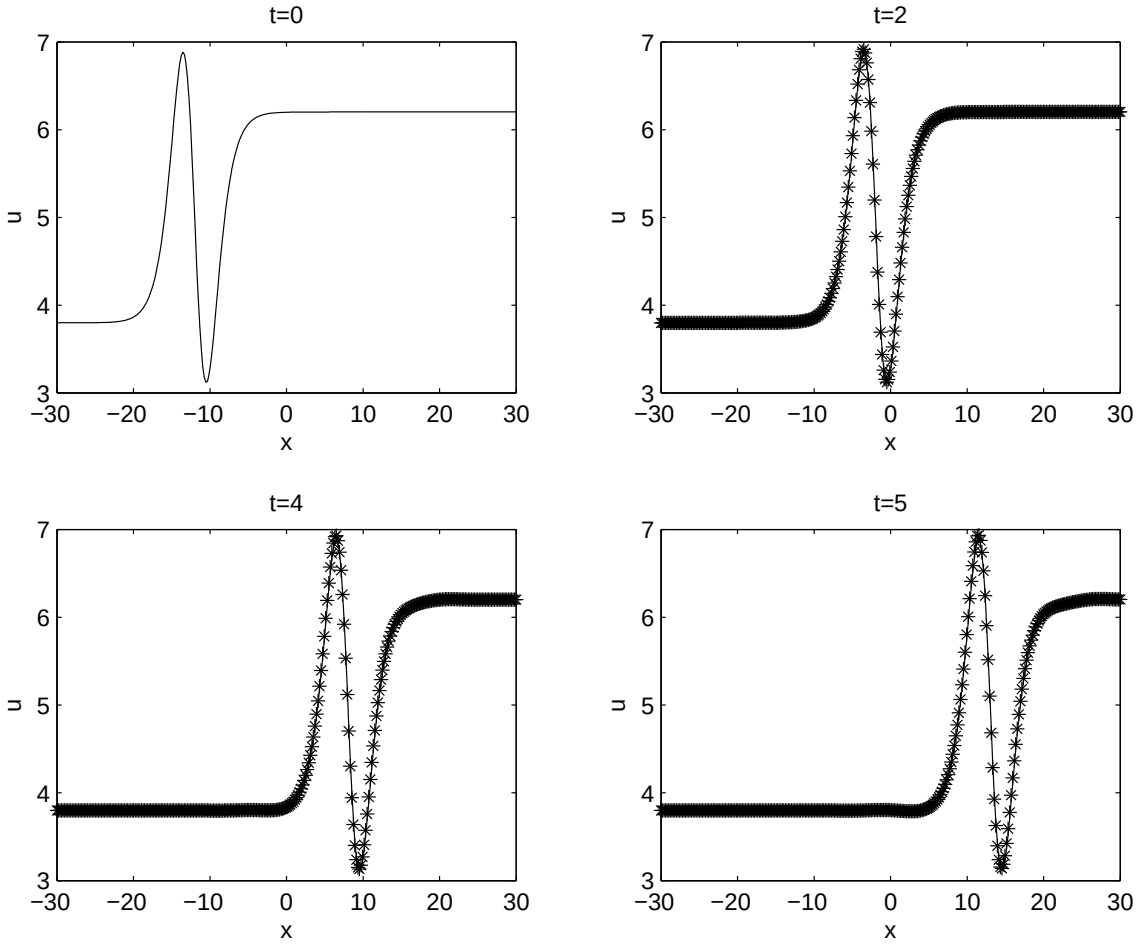

Figure 6.1: Plots of the computational result for the KS equation by scheme 6.9.

From Fig.6.1 we see our numerical scheme successfully captures the traveling wave solution of the KS equation, and from Table 6.1 we see we have a clean second order scheme in space.

Example 2: Long time simulation

We perform the numerical test for the equation

$$u_t + uu_x + u_{xx} + \sigma u_{xxx} + u_{xxxx} = 0. \tag{6.11}$$

The exact solution is

$$u(x,t) = c+9-15(tanh(k(x-ct-x_0))+tanh^2(k(x-ct-x_0))-tanh^3(k(x-ct-x_0))), \tag{6.12}$$

and initial condition is

$$u(x,t) = c + 9 - 15(tanh(k(x-x_0)) + tanh^2(k(x-x_0)) - tanh^3(k(x-x_0))), \tag{6.13}$$

where $c = 1$, $k = 1/2$, $\sigma = 4$. From (6.7) we have the numerical scheme for (6.11)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{1}{6h}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}) + D_+D_-u_j^{n+1/2} + \sigma D_0 D_+ D_- u_j^{n+1/2} + D_+D_-D_+D_-u_j^{n+1/2} = 0. \tag{6.14}$$

Table 6.2: Newmann boundary condition. Uniform meshes with N cells at time $t = 0.1$.

N	L^2	order	L^∞	order
50	5.53E-01	-	3.96E-01	-
100	1.4E-01	1.98	1.17E-01	1.77
200	3.51E-02	2.00	2.89E-02	2.01
400	8.77E-03	2.00	7.26E-03	1.99

We notice the soliton has an advection speed of 1, we take the computational domain to be $[-10, 10]$, a period of the soliton is $\frac{20}{1} = 20$, we take 200 cells $\Delta t = 0.01$ and display the soliton up to 40(2 periods), 100(5 periods) and 200(10 periods) in (6.2). We see the scheme (6.14) is statble in long-time simulation with large time step. There is also phase error when the simulation time is long due to the numerical error of the second order scheme, we can reduce the error by making finer grid cell or taking smaller time step Δt.

Example 3: Another type K.S. equation

We perform a numerical test for the equation

$$u_t + uu_x - u_{xx} + u_{xxxx} = 0, \tag{6.15}$$

with the initial condition

$$u(x,0) = c + \frac{15}{19\sqrt{19}}(-3tanh(k(x-x_0)) + tanh^3(k(x-x_0))). \tag{6.16}$$

The exact solution

$$u(x,t) = c + \frac{15}{19\sqrt{19}}(-3tanh(k(x-ct-x_0)) + tanh^3(k(x-ct-x_0))), \tag{6.17}$$

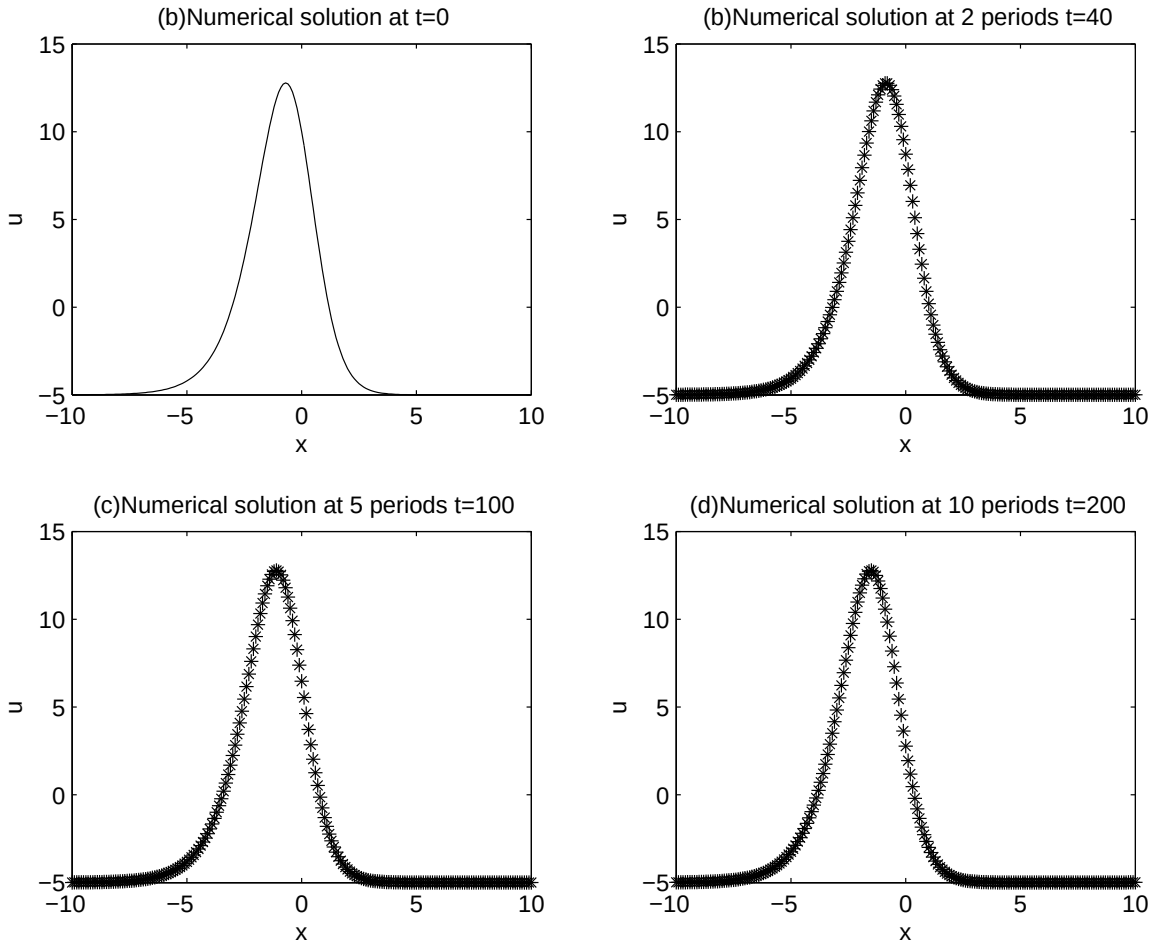

Figure 6.2: Plots of the computational result for KS equation 6.11.

where $c = 0.8, k = \frac{1}{2\sqrt{19}}, x_0 = -10$. From (6.7) we have the numerical scheme for (6.15)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{1}{6h}(u_{j+1}^{n+1/2} + u_j^{n+1/2} + u_{j-1}^{n+1/2})(u_{j+1}^{n+1/2} - u_{j-1}^{n+1/2}) - D_+D_-u_j^{n+1/2} + D_+D_-D_+D_-u_j^{n+1/2} = 0. \tag{6.18}$$

And we take Newmann boundary condition. The shock profile wave propagation for equation is displayed in Fig.6.3, we see the moving shock profile is resolved very well.

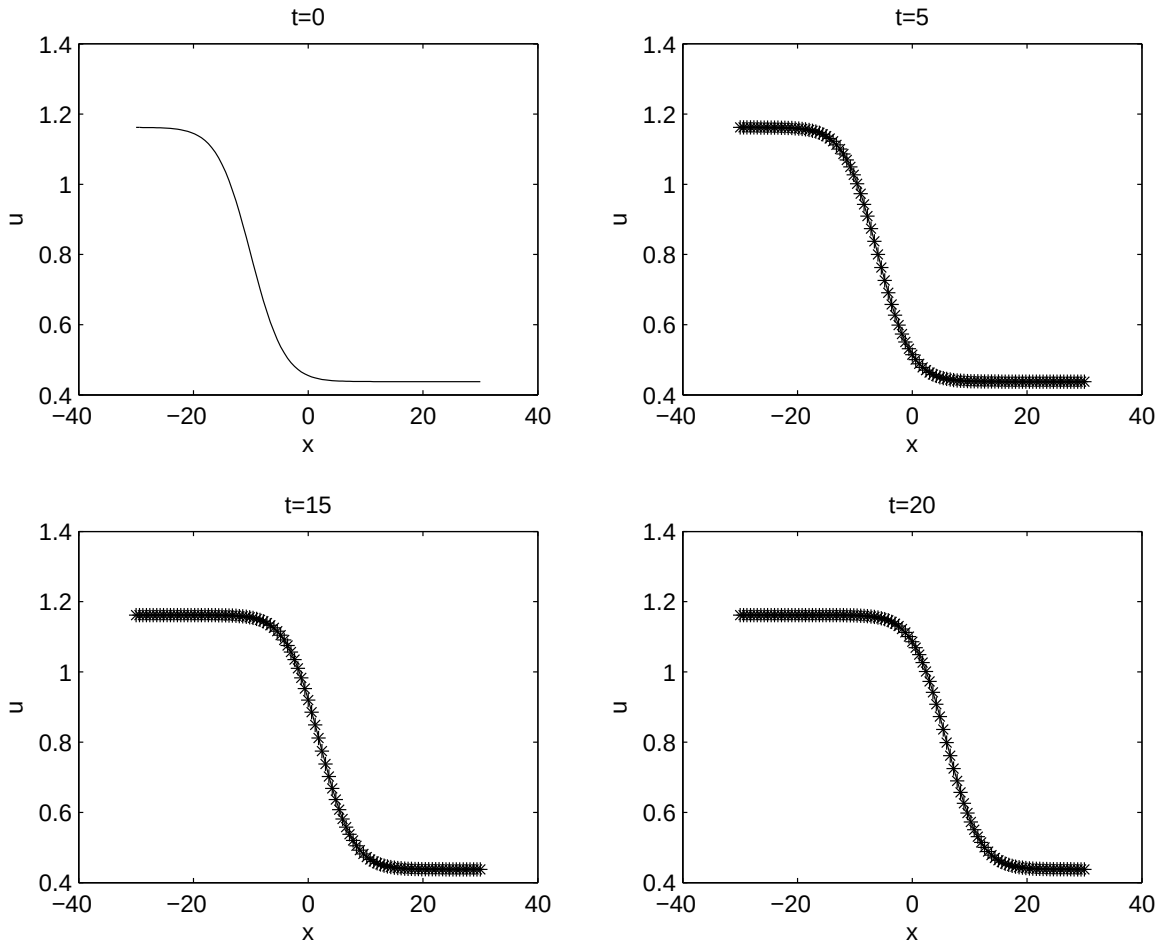

Figure 6.3: Plots of the computational result for KS equation 6.15.

6.4 Concluding remarks

In this chapter we established the L_2 stability of the finite difference scheme (6.7) for the generalized Kuramoto-Sivashinsky equation (6.2). We conduct numerical test for three different types of Kuramoto-Sivashinsky equations to show the accuracy and long time stability of scheme (6.7).

CHAPTER 7. Nonlinear model

In this chapter we consider

$$u_t = \Delta u - \nabla \cdot \Big(\frac{\nabla u}{|\nabla u|}\Big), x \in \Omega \in \mathbb{R}^2, \tag{7.1}$$

which is the gradient flow of the functional

$$I[u] = \frac{1}{2}\int_\Omega (|\nabla u| - 1)^2 dx. \tag{7.2}$$

First we will derive some energy identities of the continuous model (7.1), then we will propose two numerical schemes to keep these identities at the discrete level.

7.1 Algorithm design

Consider a rectangular region $\Omega : x\epsilon[0, L_x], y\epsilon[0, L_y]$ with boundary $\partial\Omega$ at which different types of b.c. can be imposed.

Lemma 7.1.1. *Let u(x,t) be a solution of (7.1). If one of the following admissible boundary conditions is imposed:*

$$\nabla u \cdot n = 0, |\nabla u| = 1,$$

then the energy identities

$$\frac{d}{dt} I(u) + ||u_t||^2 = 0, \tag{7.3}$$

$$\frac{d}{dt}||u||^2 + I(u) + ||u||_{L1} = |\Omega|, \tag{7.4}$$

hold, where $||.||$ is the standard L^2-norm in Ω, $||\cdot||_{L^1}$ is the standard L^1 norm, and

$$I[u] = \frac{1}{2}\int_\Omega (|\nabla u| - 1)^2 dx. \tag{7.5}$$

Proof of (7.3),(7.4) is useful in deriving their discrete counterparts. It follows from (7.1) that

$$\begin{aligned} \langle u_t, u_t\rangle - \langle \Delta u - \nabla\cdot(\frac{\nabla u}{|\nabla u|}), u_t\rangle &= 0, \\ \langle u_t, u_t\rangle + \langle (|\nabla u|-1)\frac{\nabla u}{|\nabla u|}, \nabla u_t\rangle - \int_{\partial\Omega}(|\nabla u|-1)\frac{\nabla u}{|\nabla u|}\cdot n u_t dS &= 0, \end{aligned}$$

where $\langle\cdot,\cdot\rangle$ denotes the standard inner product in the L_2 space, n denote the outward normal direction of $\partial\Omega$. We notice

$$\frac{d}{dt}I(u) = \langle (|\nabla u|-1)\frac{\nabla u}{|\nabla u|}, \nabla u_t\rangle, \tag{7.6}$$

which gives us the identity (7.3). Next we will prove identity (7.4), it follows from (7.1 that

$$\begin{aligned} \langle u_t, u\rangle - \langle \Delta u - \nabla\cdot(\frac{\nabla u}{|\nabla u|}), u\rangle &= 0, \\ \frac{d}{dt}||u||^2 + \int_\Omega (|\nabla u|^2 - 2|\nabla u| + 1 + |\nabla u|)dx &= 0, \\ \frac{d}{dt}||u||^2 + I(u) + \int_\Omega |\nabla u| dx &= |\Omega|, \end{aligned}$$

which completes the proof of the lemma.

The original equation satisfies the energy identity, so whatever numerical scheme we use, we should try to keep this property. We first discretize in time, keeping u to be continuous function of space variables to obtain a semi-discrete numerical scheme,

$$\frac{u^{n+1}-u^n}{\Delta t} = \frac{1}{2}(\Delta u^{n+1} + \Delta u^n) - \nabla\cdot\Big(\frac{\nabla u^{n+1} + \nabla u^n}{|\nabla u^{n+1}| + |\nabla u^n|}\Big), \tag{7.7}$$

where $u^n = u(x, y, t^n)$.

Lemma 7.1.2. *The scheme (7.7) satisfies the energy identity (7.3).*

Proof. We define the discrete form of (7.2) to be $I^n = \int_\Omega \frac{1}{2}\Big((|\nabla u^n|-1)^2\Big)dx$

$$\begin{aligned} & I^{n+1} - I^n \\ = & \int_\Omega \frac{1}{2}\Big((|\nabla u^{n+1}|-1)^2 - (|\nabla u^n|-1)^2\Big)dx \\ = & \int_\Omega \Big(\frac{1}{2}(|\nabla u^{n+1}|^2 - |\nabla u^n|^2) - (|\nabla u^{n+1}| - |\nabla u^n|)\Big)dx \\ = & \int_\Omega (|\nabla u^{n+1}| - |\nabla u^n|)\big(\frac{1}{2}(|\nabla u^{n+1}| + |\nabla u^n|) - 1\big)dx. \end{aligned}$$

We notice that

$$\begin{aligned} & |\nabla u^{n+1}| - |\nabla u^n| \\ = \; & \frac{(|\nabla u^{n+1}| + |\nabla u^n|)(|\nabla u^{n+1}| - |\nabla u^n|)}{(|\nabla u^{n+1}| + |\nabla u^n|)} \\ = \; & \frac{(\nabla u^{n+1} + \nabla u^n) \cdot (\nabla u^{n+1} - \nabla u^n)}{(|\nabla u^{n+1}| + |\nabla u^n|)}. \end{aligned}$$

So we have

$$\begin{aligned} & \int_\Omega (|\nabla u^{n+1}| - |\nabla u^n|)\big(\frac{1}{2}(|\nabla u^{n+1}| + |\nabla u^n|) - 1\big)dx \\ = \; & \int_\Omega (\nabla u^{n+1} + \nabla u^n) \cdot (\nabla u^{n+1} - \nabla u^n)\Big(\frac{1}{2} - \frac{1}{|\nabla u^{n+1}| + |\nabla u^n|}\Big)dx. \end{aligned} \tag{7.8}$$

By integration by parts formula, we have

$$(7.8) = -\frac{1}{2}\int_\Omega \Big(\Delta u^{n+1} + \Delta u^n - \nabla \cdot \frac{\nabla u^{n+1} + \nabla u^n}{|\nabla u^{n+1}| + |\nabla u^n|}\Big)(u^{n+1} - u^n)dx. \tag{7.9}$$

Then according to scheme (7.7) we have

$$I^{n+1} - I^n + \frac{1}{\Delta t}\int_\Omega (u^{n+1} - u^n)(u^{n+1} - u^n)dx = 0.$$

This is exactly the Energy Identity (7.3) in the discrete form. □

However, this scheme is hard to implement because the nonlinear implicit term $|\nabla u^{n+1}|$ appears in the denominator.

Next we propose a θ scheme which is easier to implement

$$\frac{u^{n+1} - u^n}{\Delta t} - \theta \Delta u^{n+1} = (1-\theta)\Delta u^n - \nabla \cdot (\frac{\nabla u^n}{|\nabla u^n|}). \tag{7.10}$$

This scheme is consistent with the original equation for $0 < \theta \leq 1$. Next we choose the value of θ to preserve the energy property.

Lemma 7.1.3. *If $\frac{1}{2} \leq \theta \leq 1$, then the θ scheme preserves the energy decreasing property at the discrete level, i.e, $I^{n+1} \leq I^n$.*

Proof. Set $\Phi = u^{n+1} - u^n$. It follows from (7.10) that

$$\begin{aligned}
\frac{1}{\Delta t}\langle u^{n+1} - u^n, \Phi\rangle - \langle \theta\Delta u^{n+1} + (1-\theta)\Delta u^n, \Phi\rangle + \langle \nabla\cdot(\frac{\nabla u_n}{|\nabla u_n|}), \Phi\rangle &= 0, \\
\frac{1}{\Delta t}||u^{n+1} - u^n||^2 + I^{n+1} - I^n - (\frac{1}{2}|||\nabla u^{n+1}| - 1||^2 - \frac{1}{2}|||\nabla u^n| - 1||^2) & \\
+\langle \theta\nabla u^{n+1} + (1-\theta)\nabla u^n, \nabla\Phi\rangle - \langle \frac{\nabla u_n}{|\nabla u_n|}, \nabla\Phi\rangle &= 0.
\end{aligned}$$

We let $a = \nabla u^{n+1}$,$b = \nabla u^n$, from the previous identity we have

$$\begin{aligned}
& -(\frac{1}{2}|||\nabla u^{n+1}| - 1||^2 - \frac{1}{2}|||\nabla u^n| - 1||^2) + \langle \theta\nabla u^{n+1} + (1-\theta)\nabla u^n, \nabla\Phi\rangle - \langle \frac{\nabla u_n}{|\nabla u_n|}, \nabla\Phi\rangle \\
= \; & -\frac{1}{2}(|a|-1)^2 + \frac{1}{2}(|b|-1)^2 + \langle \theta a + (1-\theta)b, a-b\rangle - \langle \frac{b}{|b|}, a-b\rangle \\
= \; & -\frac{1}{2}(|a|-1)^2 + \frac{1}{2}(|b|-1)^2 + \theta|a|^2 - (1-\theta)|b|^2 - \theta a\cdot b + (1-\theta)|b|^2 - \theta a\cdot b \\
& +(1-\theta)b\cdot a - \frac{b}{|b|}\cdot(a-b) \\
= \; & (\theta - \frac{1}{2})|a-b|^2 + \frac{|a||b| - b\cdot a}{|b|} \geq 0
\end{aligned}$$

According to Cauchy-Schwartz inequality $|a||b| \geq b\cdot a$, we only need the first term to be positive, we see $\frac{1}{2} \leq \theta \leq 1$, which finishs the proof. □

Remark: Actually $(\theta - \frac{1}{2})|a-b|^2 + \frac{|a||b|-b\cdot a}{|b|}$is the additional term compared to the continuous energy identity, So we need to keep θ as small as possible, so we take $\theta = \frac{1}{2}$.

7.2 Numerical Examples

In one dimensional space (7.1) reduces to equation

$$\partial_t u = u_{xx} - \partial_x(\frac{\partial_x u}{|\partial_x u|}). \tag{7.11}$$

First we discretize (7.11) in time, it follows from (7.10)

$$(I - \Delta t\partial_{xx})u^{n+1} = u^n + \Delta t(1-\theta)\partial_{xx}u^n - \Delta t\partial_x(\frac{\partial_x u^n}{|\partial_x u^n|}) \tag{7.12}$$

For spatial discretization we use central difference approximation

$$\begin{aligned}
\partial_{xx}u_j &\approx \frac{1}{(\Delta x)^2}(u_{j+1} - 2u_j + u_{j-1}), \\
\partial_x(\frac{\partial_x u^n}{|\partial_x u^n|}) &\approx \frac{1}{\Delta x}(\frac{u_{j+1}-u_j}{|u_{j+1}-u_j|} - \frac{u_j - u_{j-1}}{|u_j - u_{j-1}|}).
\end{aligned}$$

We choose the initial function $\phi_0 = 1-x^2$, and use the Newman boundary condition(constant extension) which guarantees $\nabla u = 0$ on the boundary. The final steady state solution is displayed in figure 7.1.The numerical result shows we have a steady state solution $|\nabla u| = 1$, which is a signed distance function plus a constant.

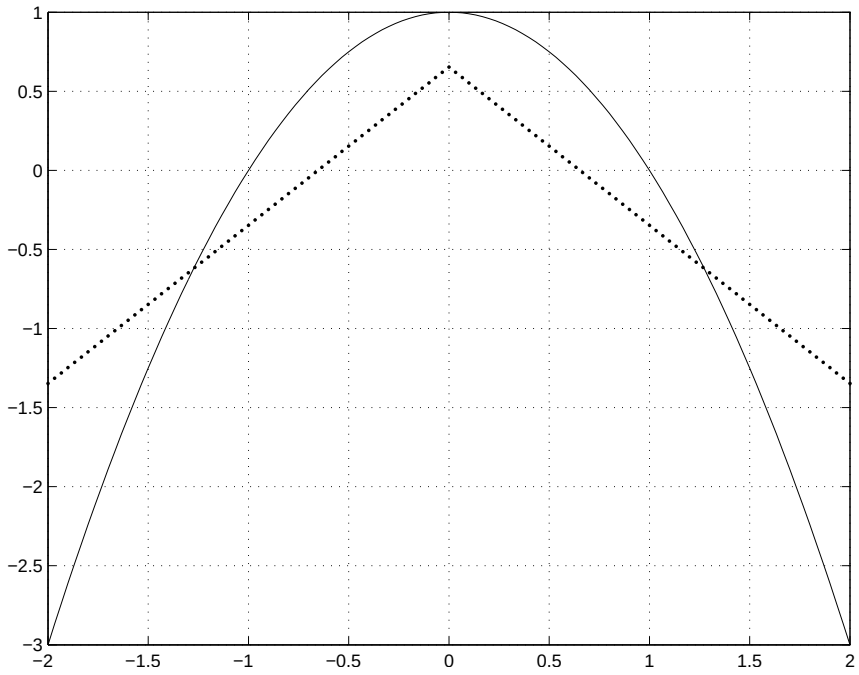

Figure 7.1: Plots of the computational result for model equation 7.11.

Further more, we have tried different initial functions $\phi_0 = x^2$,$\phi_0 = x(1-x^2)$, and always get the expected solution $|\nabla u| = 1$.

7.3 2D Implementation

In order to implement the θ scheme in two-dimensional space we use ADI scheme to split the operator, it is follows from (7.10)

$$(I - \Delta t\theta\Delta)u^{n+1} = u^n + \Delta t(1-\theta)\Delta u^n - \Delta t\nabla \cdot (\frac{\nabla u^n}{|\nabla u^n|}), \tag{7.13}$$

We define

$$L_x = I - \theta\Delta t\Delta\partial_{xx}, L_y = I - \theta\Delta t\partial_{yy}.$$

By this definition we have

$$(I - \Delta t\theta\Delta) = L_x L_y - \theta^2 \Delta t^2 \partial_{xx}\partial_{yy},$$

then

$$(L_x L_y - \theta^2 \Delta t^2 \partial_{xx}\partial_{yy})u^{n+1} = (I + \Delta t(1-\theta)\Delta)u^n - \nabla \cdot (\frac{\Delta u^n}{|\Delta u^n|})\Delta t.$$

Notice the term $\theta^2 \Delta t^2 \partial_{xx}\partial_{yy}$is of order $O(\Delta t^2)$, without loss of accuracy we can apply this term to u^ninstead of u^{n+1} to abtain

$$L_x L_y u^{n+1} = (I + \Delta t(1-\theta)\Delta)u^n + \theta^2 \Delta t^2 \partial_{xx}\partial_{yy}u^n - \nabla \cdot (\frac{\nabla u^n}{|\nabla u^n|})\Delta t,$$

while introducing only high order correction $o(\Delta t^3)$. Substract $L_x L_y u^n$both sides and define $v = u^{n+1} - u^n$then

$$L_x L_y v = \Delta t \Delta u^n - \nabla \cdot (\frac{\nabla u^n}{|\nabla u^n|})\Delta t,$$

with the generated ADI operator-split form

$$\begin{aligned} L_x w &= \Delta t \Delta u^n - \nabla \cdot \frac{(\nabla u^n)}{|\nabla u^n|}\Delta t, \\ L_y v &= w, \\ u^{n+1} &= u^n + v. \end{aligned}$$

Next we discuss the way to discretize $\nabla \cdot (\frac{u^n}{|u^n|})$.

$$\begin{aligned} \nabla \cdot (\frac{u^n}{|u^n|}) &= \frac{A_{i+1,j} - Ai,j}{\Delta x} - \frac{B_{i,j+1} - B_{i,j}}{\Delta y}, \\ A_{i,j} &= C_{i-\frac{1}{2},j}(u_{i,j} - u_{i-1,j}), \\ B_{i,j} &= D_{i,j-\frac{1}{2}}(u_{i,j} - u_{i,j-1}), \\ C_{i-\frac{1}{2},j} &= \frac{1}{\sqrt{\Delta_x^- u_{i,j} + (\frac{\Delta_x^0 u_{i,j-1}}{2})^2 + (\frac{\Delta_x^0 u_{i,j}}{2})^2 + \delta\Delta y^2}}, \\ D_(i, j-\frac{1}{2}) &= \frac{1}{\sqrt{\Delta_y^- u_{i,j} + (\frac{\Delta_x^0 u_{i,j-1}}{2})^2 + \frac{\Delta_x^0 u_{i,j}}{2} + \delta\Delta y^2}}. \end{aligned}$$

7.4 conclusion

In this chapter we have designed two large time step schemes (7.7),(7.10)for the model equation (7.1), the implementation of (7.10) in one dimension give us the desired solution

$|\nabla u| = 1$. For two dimensional problem we give the numerical algorithm by applying the dimension splitting technique.

BIBLIOGRAPHY

[1] G. D. Akrivis. Finite difference discretization of the Kuramoto-Sivashinsky equation. *Numer. Math.*, 63(1):1–11, 1992.

[2] U. M. Ascher and R. I. McLachlan. Multisymplectic box schemes and the Korteweg-de Vries equation. *Appl. Numer. Math.*, 48(3-4):255–269, 2004. Workshop on Innovative Time Integrators for PDEs.

[3] U. M. Ascher and R. I. McLachlan. On symplectic and multisymplectic schemes for the KdV equation. *J. Sci. Comput.*, 25(1-2):83–104, 2005.

[4] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton. Implicit-explicit methods for time-dependent partial differential equations. *SIAM J. Numer. Anal.*, 32(3):797–823, 1995.

[5] C. J. Budd and R. Kuske. Localized periodic patterns for the non-symmetric generalized Swift-Hohenberg equation. *Phys. D*, 208(1-2):73–95, 2005.

[6] T. F. Chan and T. Kerkhoven. Fourier methods with extended stability intervals for the Korteweg-de Vries equation. *SIAM J. Numer. Anal.*, 22(3):441–454, 1985.

[7] C. I. Christov and J. Pontes. Numerical scheme for Swift-Hohenberg equation with strict implementation of Lyapunov functional. *Math. Comput. Modelling*, 35(1-2):87–99, 2002.

[8] C. I. Christov, J. Pontes, D. Walgraef, and M. G. Velarde. Implicit time splitting for fourth-order parabolic equations. *Comput. Methods Appl. Mech. Engrg.*, 148(3-4):209–224, 1997.

[9] S. M. Cox and P. C. Matthews. Exponential time differencing for stiff systems. *J. Comput. Phys.*, 176(2):430–455, 2002.

[10] Y. F. Cui and D. K. Mao. A difference scheme satisfying two conservation laws for a KdV equation. *Comm. Appl. Math. Comput.*, 19(2):15–22, 2005.

[11] T. A. Driscoll. A composite Runge-Kutta method for the spectral solution of semilinear PDEs. *J. Comput. Phys.*, 182(2):357–367, 2002.

[12] B. Fornberg and T. A. Driscoll. A fast spectral algorithm for nonlinear wave equations with linear dispersion. *J. Comput. Phys.*, 155(2):456–467, 1999.

[13] B. Gustafsson, H.-O. Kreiss, and J. Oliger. *Time dependent problems and difference methods*. Pure and Applied Mathematics (New York). John Wiley & Sons Inc., New York, 1995. A Wiley-Interscience Publication.

[14] Y. He, Y. Liu, and T. Tang. On large time-stepping methods for the Cahn-Hilliard equation. *Appl. Numer. Math.*, 57(5-7):616–628, 2007.

[15] P. C. Hohenberg and J. B. Swift. Effects of additive noise at the onset of rayleigh-bénard convection. *Phys. Rev. A*, 46(8):4773–4785, Oct 1992.

[16] H. Holden, K. H. Karlsen, and N. H. Risebro. Operator splitting methods for generalized Korteweg-de Vries equations. *J. Comput. Phys.*, 153(1):203–222, 1999.

[17] H. Holden, K. H. Karlsen, and N. H. Risebro. Operator splitting methods for generalized Korteweg-de Vries equations. *J. Comput. Phys.*, 153(1):203–222, 1999.

[18] A. P. Hooper and R. Grimshaw. Travelling wave solutions of the Kuramoto-Sivashinsky equation. *Wave Motion*, 10(5):405–420, 1988.

[19] J. M. Hyman and B. Nicolaenko. The Kuramoto-Sivashinsky equation: a bridge between PDEs and dynamical systems. *Phys. D*, 18(1-3):113–126, 1986. Solitons and coherent structures (Santa Barbara, Calif., 1985).

[20] G. E. Karniadakis, M. Israeli, and S. A. Orszag. High-order splitting methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 97(2):414–443, 1991.

[21] A.-K. Kassam and L. N. Trefethen. Fourth-order time-stepping for stiff PDEs. *SIAM J. Sci. Comput.*, 26(4):1214–1233 (electronic), 2005.

[22] J. Kim and P. Moin. Application of a fractional-step method to incompressible Navier-Stokes equations. *J. Comput. Phys.*, 59(2):308–323, 1985.

[23] J. Lega, J. V. Moloney, and A. C. Newell. Swift-hohenberg equation for lasers. *Phys. Rev. Lett.*, 73(22):2978–2981, Nov 1994.

[24] C. D. Levermore and J.-G. Liu. Large oscillations arising in a dispersive numerical scheme. *Phys. D*, 99(2-3):191–216, 1996.

[25] D. Michelson. Steady solutions of the Kuramoto-Sivashinsky equation. *Phys. D*, 19(1):89–111, 1986.

[26] K. W. Morton and D. F. Mayers. *Numerical solution of partial differential equations*. Cambridge University Press, Cambridge, second edition, 2005. An introduction.

[27] L. A. Peletier and V. Rottschäfer. Pattern selection of solutions of the Swift-Hohenberg equation. *Phys. D*, 194(1-2):95–126, 2004.

[28] L. A. Peletier and W. C. Troy. Pattern formation described by the Swift-Hohenberg equation. *Sūrikaisekikenkyūsho Kōkyūroku*, (1178):1–15, 2000. Nonlinear diffusive systems—dynamics and asymptotic analysis (Japanese) (Kyoto, 2000).

[29] B. Peña, C. Perez-Garcia, and B. Echebarria. Stability of hexagonal patterns in a generalized Swift-Hohenberg equation. In *Space-time chaos (Pamplona, 2000)*, pages 123–136. World Sci. Publ., River Edge, NJ, 2001.

[30] S. J. Ruuth. Implicit-explicit methods for reaction-diffusion problems in pattern formation. *J. Math. Biol.*, 34(2):148–176, 1995.

[31] J. Swift and P. C. Hohenberg. Hydrodynamic fluctuations at the convective instability. *Phys. Rev. A*, 15(1):319–328, Jan 1977.

[32] E. Tadmor. The well-posedness of the Kuramoto-Sivashinsky equation. *SIAM J. Math. Anal.*, 17(4):884–893, 1986.

[33] L. N. Trefethen. *Spectral methods in MATLAB*, volume 10 of *Software, Environments, and Tools*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[34] C. Xu and T. Tang. Stability analysis of large time-stepping methods for epitaxial growth models. *SIAM J. Numer. Anal.*, 44(4):1759–1779 (electronic), 2006.

[35] Y. Xu and C.-w. Shu. Local discontinuous Galerkin methods for three classes of nonlinear wave equations. *J. Comput. Math.*, 22(2):250–274, 2004. Special issue dedicated to the 70th birthday of Professor Zhong-Ci Shi.

[36] Y. Xu and C.-W. Shu. Local discontinuous Galerkin methods for the Kuramoto-Sivashinsky equations and the Ito-type coupled KdV equations. *Comput. Methods Appl. Mech. Engrg.*, 195(25-28):3430–3447, 2006.

[37] T.-S. Yang. On traveling-wave solutions of the Kuramoto-Sivashinsky equation. *Phys. D*, 110(1-2):25–42, 1997.

[38] M. Yari. Attractor bifurcation and final patterns of the n-dimensional and generalized Swift-Hohenberg equations. *Discrete Contin. Dyn. Syst. Ser. B*, 7(2):441–456 (electronic), 2007.

[39] P. F. Zhao and M. Z. Qin. Multisymplectic geometry and multisymplectic Preissmann scheme for the KdV equation. *J. Phys. A*, 33(18):3613–3626, 2000.